AMBIENTE DE INTEGRAÇÃO EM EQUIPES DE FÁBRICA DE SOFTWARE

Jonas XAVIER (1); Amanda FERREIRA (2); Thayssa LACERDA (3); Anderson MOREIRA (4)

Instituto Federal de Pernambuco, Coordenação de Sistemas de Informação, Av. Prof. Luiz Freire, 500 – Cidade Universitária, CEP: 50740-540, Recife - PE, +55 81 2125 1631, (1) jagx@dase.recife.ifpe.edu.br, (2) arsf3@dase.recife.ifpe.edu.br, (3) tmsl@dase.recife.ifpe.edu.br, (4) anderson.moreira@recife.ifpe.edu.br.

RESUMO

Este artigo apresenta o trabalho de Iniciação Científica aplicada no desenvolvimento de uma série de procedimentos, reunidos em um guia, que auxiliam a inclusão de desenvolvedores dentro de um ambiente de fábrica de *software*. Além de mostrar os meios possíveis para disseminação da informação no ambiente. O trabalho foi feito para auxiliar a sanar a dificuldade encontrada pelos novos funcionários/integrantes na realização e aprendizado de suas atividades em um projeto de desenvolvimento de *software.* Isso porque existe uma demanda rápida para adaptação deste funcionário ao ambiente de desenvolvimento, principalmente na fase inicial de seu vínculo ao projeto. Para auxiliar na análise da aplicabilidade do trabalho, foram utilizados dois projetos como estudo de caso: o Módulo Gerencial do Sistema de Informações da Educação Profissional (SIEP-Gerencial), projeto fomentado pelo Ministério da Educação (MEC) e a Diretoria de Avaliação e Desenvolvimento de Tecnologia da Informação (DADT), departamento vinculado à Reitoria do Instituto Federal de Pernambuco (IFPE).

Palavras-chave: trabalho colaborativo, desenvolvimento distribuído, guia de desenvolvimento

1. INTRODUÇÃO

Investir em um guia de desenvolvedor permite a integração de novos membros de forma eficaz, para que os resultados sejam mais rápidos e sem prejuízos na produção. Ambientes de fábrica de *software* demandam por pessoas especializadas em uma determina área do conhecimento e um determinado produto de software é construído conforme uma linha de produção. Porém a saída de uma pessoa nessa linha de produção de software pode ser danosa ao gerenciamento do projeto. Técnicas que envolvem Gerência de Projeto como o PMP, auxiliam a sanar esse problema, porém modelos fáceis que auxiliam no aprendizado de determinada tecnologia são difíceis de encontrar.

Como base, toma-se por exemplos o SIEP-GERENCIAL, projeto que trabalha de acordo das demandas do MEC, e o DADT, diretoria de tecnologia do IFPE, em ambos, os sistemas em desenvolvimento precisam ter flexibilidade e agilidade para que possam aderir às diversas realidades em que serão implantados. Tanto o afastamento como a integração de membros nas equipes são acontecimentos comuns em um projeto, o que compromete o sincronismo no ritmo de trabalho, este que deve ser mantido para não comprometer a produtividade. Nesses dois projetos a grande mão de obra de trabalho são alunos bolsistas, e a rotatividade destes é muito grande devido a uma série de fatores, entre eles formatura, novas oportunidades de trabalho fora do Instituto, entre outras.

Consideráveis formas de integração foram selecionadas, entre elas o guia do desenvolvedor se mostrou a mais eficiente, destacando-se sua interatividade com o usuário. O SIEP-GERENCIAL e o DADT fazem uso dessa forma de integração com seus novos membros, além de alternativas, como a alocação de um desenvolvedor mais experiente para treinamento, o que possibilitou o comparativo entre as metodologias de integração na prática, nesses ambientes.

O presente trabalho de Iniciação Científica, mostrado nesse artigo e financiado pela Fundação de Amparo a Pesquisa de Pernambuco (FACEPE) e pelo DADT do IFPE, tem como ideal facilitar a inclusão de novos desenvolvedores em uma equipe de desenvolvimento de *software*. Para isso, além das boas práticas de gerenciamento que já são conhecidas, também deve ser utilizado um ambiente que irá centralizar as informações necessárias para inclusão do desenvolvedor.

O resultado mostra que um ambiente que possui um sistema unificado de disseminação de informação é mais produtivo e pró-ativo do que um ambiente que não segue esse padrão.

Este documento está estruturado nos seguintes tópicos: na Seção 2 explica o Estado da Arte, assim como uma síntese sobre os ambientes de desenvolvimento que tomamos para exemplo, SIEP-Gerencial e DADT, e a Gerência de Conhecimento; na Seção 3 são exploradas as metodologias mais usadas e sua implantação nas fábricas de software estudadas; na Seção 4 são expostas as interfaces do Guia do Desenvolvedor e Guia do Usuário, usados no SIEP-Gerencial e DADT, respectivamente; na Seção 5 mostra os materiais utilizados como ferramentas de integração; na Seção 6 é apresentado o comparativo dos ambientes estudados; e na Seção 7 são as conclusões e trabalhos futuros.

2. ESTADO DA ARTE

O termo *software factory* (fábrica de software em inglês) foi empregado pela primeira vez no ano de 1969, pela empresa japonesa Hitachi, ganhando popularidade no início dos anos 90. A idéia era aplicar conceitos de produção industrial em ambientes de desenvolvimento de software, para aumentar a produtividade e diminuir prazos e custos (Cusumano, 1989).

"Agora nós temos computadores gigantescos, programar tornou-se um problema igualmente gigantesco" (Dijkstra, 1972). O desenvolvimento de software possui uma aflição crônica,causada pelo aumento constante da complexidade e do tamanho do sistemas criados (Pressman, 2004). Dois fatores: a criatividade e a organização, respectivamente críticos na produção e manutenção de sistemas. A criatividade está ligada às bases da computação como ciência, este campo varia de um indivíduo para outro. A organização por outro lado é necessária e pode ser praticada por todos, essa é a área que tratada pela engenharia de software (Fuggetta, 2000).

Este artigo mostra como técnicas específicas de integração são importantes em uma equipe de desenvolvimento. Equipes que trabalham com essa metodologia devem ter em mente o quanto é importante a rapidez de aprendizado de seus membros. Entre os benefícios: dinamismo, segurança ao projeto, descentralizando e fortalecendo aspectos fundamentais.

2.1 Sistema de Informação da Educação Tecnológica e Profissional (SIEP)

A Secretaria de Educação Profissional e Tecnologia do Ministério da Educação (SETEC) tem como responsabilidades planejar, orientar, coordenar e supervisionar processos de formulação e implementação da política de educação profissional e tecnológica e cuidar do cumprimento da legislação educacional.

De acordo com MOREIRA *et al.*, 2009, a SETEC mantém poucos dados sobre a Educação Profissional e Tecnológica (EPT) no país, principalmente sobre evolução das matrículas e perfil sócio-econômico dos alunos. Dessa forma, encontra dificuldades de comando (gestão) em várias ordens (dados, boas práticas, políticas de inclusão, etc.). Tendo esses pontos como base de trabalho a SETEC criou o SIEP para suprir as necessidades de informação do MEC/SETEC, dentro dos processos de planejamento estratégico e operacional.

Sendo dividido em duas partes, o SIEP gerencial e o SIGA EPT. Também de acordo com MOREIRA, *et al.* 2009, o SIEP-Gerencial foi projetado e desenvolvido para permitir as mais variadas formas de acesso - voltadas a atender as demandas atuais e futuras. Possibilitando consulta, monitoramento, avaliação, cruzamento e uniformidade de tratamento de dados e informações, disponibilizando saídas como relatórios, gráficos, tabelas customizáveis e integração com outros sistemas do MEC, de maneira a atender ao máximo possível a demanda de seus usuários finais.

Ficou decido que o sistema seria feito de forma colaborativa por várias instituições: IFPE, IFMA, IFAM, IFRJ, IFPB, IFRN e IFMT. Atendendo as normas do Governo federal, todo o projeto foi desenvolvido a partir de tecnologias livres. Após ser dividido em módulos, cada instituto ficou responsável por um módulo. Além disso, as equipes deveriam manter comunicação constante com outras equipes, formadas sob as mesmas circunstancias espalhadas no Brasil inteiro. A mão de obra especializada do projeto são os professores e alunos das Instituições participantes, assumindo as funções respectivamente de Gerentes, e desenvolvedores.

2.2 Diretoria de Análise e Desenvolvimento em Tecnologias

O DADT foi criado pelo IFPE com o intuito de desenvolver e analisar novos projetos tecnológicos. Os alunos estagiários do Instituto atuam no núcleo de desenvolvimento através das demandas decorrentes a reitoria, independente do campus que a tenha requisitado. Os projetos são totalmente ligados às novas tecnologias, como produção e testes de softwares.

O principal papel da DADT é garantir que o papel das diretorias de tecnologias dos campi estejam atuando de forma correta e que sigam determinados padrões de organização. Com isso facilita no trabalho estratégico de planejamento e gestão.

2.3. Gerência de Conhecimento

Empresas que atuam diretamente com tecnologia devem ter muito cuidado com as informações obtidas no decorrer do dia-a-dia, principalmente para que o conhecimento fique retido, em algumas situações, dentro da empresa. De acordo com STAIR e REYNOLDS, 2009, a informação e o conhecimento gerado dentro das empresas são bens de importância maior ou igual aos recursos financeiros, não podendo ser disseminada de qualquer forma ou a esmo sem um prévio conhecimento das consequências que isso possa gerar. Para demonstrar a gerencia do conhecimento podemos citar o núcleo de desenvolvimento do SIEP-Gerencial. Em que foi desenvolvido um processo de gerenciamento e gestão das informações. O Núcleo não possuía um processo de captura e dissipação do conhecimento, muitas vezes, membros recém contratado demoravam muito até alcançar um bom nível de produtividade, pois o repasse das informações internas demandava muito tempo. Vamos apontar alguns tópicos chaves identificados, a seguir, com a perspectiva do ramo de desenvolvimento de *software*:

- Motivação – Encorajar os desenvolvedores a contribuir, pois dessa forma tornará o conhecimento algo rico e de fácil acesso;
- Captura do conhecimento – Tornar fácil e viável a habilidade de capturar os conhecimentos importantes para o núcleo;
- Armazenamento de conhecimento - Criação de um repositório único de conhecimentos adquiridos no cotidiano dos integrantes da equipe;
- Treinamento – Métodos elaborados para melhor treinamento do recém-contratado;
- Transferência de conhecimento - Efetividade do compartilhamento de conhecimentos a partir do fácil acesso a algum recurso computacional.

Estes cinco pontos foram essenciais para confecção do Guia do Desenvolvedor e a implantação do sistema de gestão de conhecimento em vários aspectos dentro do núcleo. Tais tópicos ajudam os desenvolvedores do núcleo na identificação dos conhecimentos desenvolvidos individualmente tornando-o disponível a qualquer momento, sendo assim, reutilizar o *knowhow* e a experiência de cada integrante para os demais. Essa experiência incentiva a cooperação, o compartilhamento e o aprendizado continuo na equipe. (HERBSLEB *et al.* 2000), (EBERT, 2006), (MOITRA, 2001).

3. METODOLOGIA

A metodologia aqui apresentada foi fundamentada levando em consideração todo o ciclo de desenvolvimento da ideia desde sua pesquisa, passando por elaboração até chegar ao produto final e implantação. O trabalho foi dividido em cinco fases elas estão descritas abaixo:

3.1 Análises de Metodologias de Aprendizagem em Ambiente de Trabalho

Inicialmente foi desenvolvida uma pesquisa para evidenciar as metodologias que poderiam ser utilizadas para elaboração de treinamentos no ambiente de trabalho.

Um levantamento foi efetuado com o objetivo de conhecer quais as metodologias mais eficazes e as de menor custo para oferecer a inserção de um novo integrante à equipe e no acesso a informação entre os desenvolvedores já agregados a mesma. As metodologias que mais se destacaram foram:

Alocação de Desenvolvedor Experiente

Esta metodologia é efetuada com a alocação de um dos desenvolvedores da equipe para o treinamento do novo integrante, este que ministrará o treinamento vai apresentar todas as instâncias da equipe, os papéis e em algumas vezes treinar o novo integrante nas ferramentas e tecnologias que serão por ele utilizadas no seu dia-a-dia de trabalho. Apresenta como ponto positivo a facilidade de aprendizado, pois o novo integrante será acompanhado por um desenvolvedor experiente, que está ligado diretamente à produção, fazendo com que o mesmo entenda com mais precisão a real necessidade da equipe. Em contra partida, o ponto negativo é que a alocação de um integrante da equipe para o treinamento acarretaria na diminuição da produtividade do projeto, visto que seria retirada mão de obra qualificada e experiente pra outra atividade.

Distribuição de Desenvolvedores (Equipe) para Treinamento

Nesta metodologia a equipe inteira participará do treinamento do novo integrante. Os desenvolvedores efetuam um *workshop* de apresentação de conteúdo onde serão mostradas todas as ferramentas, papéis e atribuições. Em um determinado momento, cada um desses desenvolvedores estará em contato com o mesmo para oferecer um suporte mais específico sobre o produto em questão. Com isso diminui o tempo de alocação do integrante específico, mas ainda teríamos um custo na produção, pois, nesses determinados momentos, o integrante da equipe estaria alocado para o treinamento ao invés de estar desenvolvendo suas atividades.

Disponibilização de Procedimentos

Nesta metodologia, cada integrante da equipe no decorrer de seu trabalho, cria procedimentos, ou seja, uma documentação relatando os detalhes de suas atividades. Quando um novo integrante é anexado ao projeto, faz-se uso destes procedimentos para se atualizar das atividades da equipe e nas tecnologias utilizadas. Neste caso, não seria necessária a alocação de um desenvolvedor da equipe diretamente para o treinamento, entretanto, teríamos um sequencia de informações não necessariamente organizadas e compiladas que nem sempre são satisfatórias para uma aprendizagem rápida e eficiente.

3.2. Implantação da Metodologia Escolhida no SIEP-Gerencial

Baseado nos parâmetros iniciais que obtivemos na pesquisa acima decidimos utilizar no SIEP-Gerencial a *metodologia de procedimentos*, pois o núcleo possui uma grande massa de informação disponibilizada neste formato. O novo desenvolvedor teria acesso facilitado ao conhecimento sempre que fosse necessário sem ter que desviar nenhum desenvolvedor da produção para o treinamento. Na confecção dos procedimentos, observamos que a disposição de imagens das telas de instalação, ou manuseio dos sistemas, inseria melhor o novo integrante no ambiente de desenvolvimento, pois a simples exposição de informações em formato texto torna o aprendizado mais cansativo. Foi utilizado um ambiente *Web* no formato *wiki* para disponibilizar essas informações de forma mais iterativa, onde o usuário poderia acessar e também alterar o conteúdo de acordo com suas atividades e aprendizados.

3.3. Implantação da Metodologia Escolhida no DADT

A metodologia escolhida no DADT foi a de alocar um membro da equipe já experiente para treinamento. Nesse caso é mais fácil o aprendizado, pois o DADT possui vários projetos, e o novo integrante só iria focar no projeto ao qual foi designado. Utilizamos a ferramenta Guia do Usuário do *Zoho Project* (ZOHO, 2009) que se assemelha ao Guia do Desenvolvedor, para disponibilizar as informações sobre cada projeto, assim como o calendário contendo os prazos e as atividades a serem cumpridas.

4. AMBIENTES DE APRENDIZADO

O Guia do Desenvolvedor, trata-se de um ambiente *Web*, criado no presente trabalho, onde cada desenvolvedor será capaz de consultar e cadastrar informações e procedimentos. O objetivo desse sistema é facilitar o trabalho dos desenvolvedores e manter as informações publicadas nele acessíveis a todos. O Guia do Desenvolvedor foi desenvolvido em cima da plataforma *JAVA* e é a principal ferramenta de gestão de conhecimento no núcleo de Pernambuco do SIEP-Gerencial. Com a implantação desse sistema, verificou-se uma grande melhoria na produtividade dos desenvolvedores.

4.1 Interface de Aprendizado

Na Figura 1 mostra o *layout* da página inicial do Guia do Desenvolvedor, aplicado no SIEP.

Figura 1 – Versão final do Guia do Desenvolvedor.

O sistema consiste basicamente em quatro funcionalidades: cadastrar e consultar tecnologia, cadastrar e consultar procedimento. Para o sistema, a tecnologia e o procedimento tratam-se de uma página em HTML com as informações textuais e gráficas utilizadas. A funcionalidade de cadastrar tecnologia sempre implica no cadastro de um procedimento, pois a diferença entre eles, é que quando cadastramos uma tecnologia iremos abordar superficialmente sobre o que ela vem a ser e como ela está sendo usada no projeto, sem a preocupação de especificar detalhadamente conhecimentos técnicos. Já o cadastro de procedimentos traz uma abordagem mais detalhada da tecnologia, por exemplo: como instalar, como fazer um protótipo funcional dela, como usar em determinadas situações e etc. Na Figura 2 mostra o *layout* da página inicial do Guia do Usuário, também criado durante o presente trabalho e aplicado no DADT.

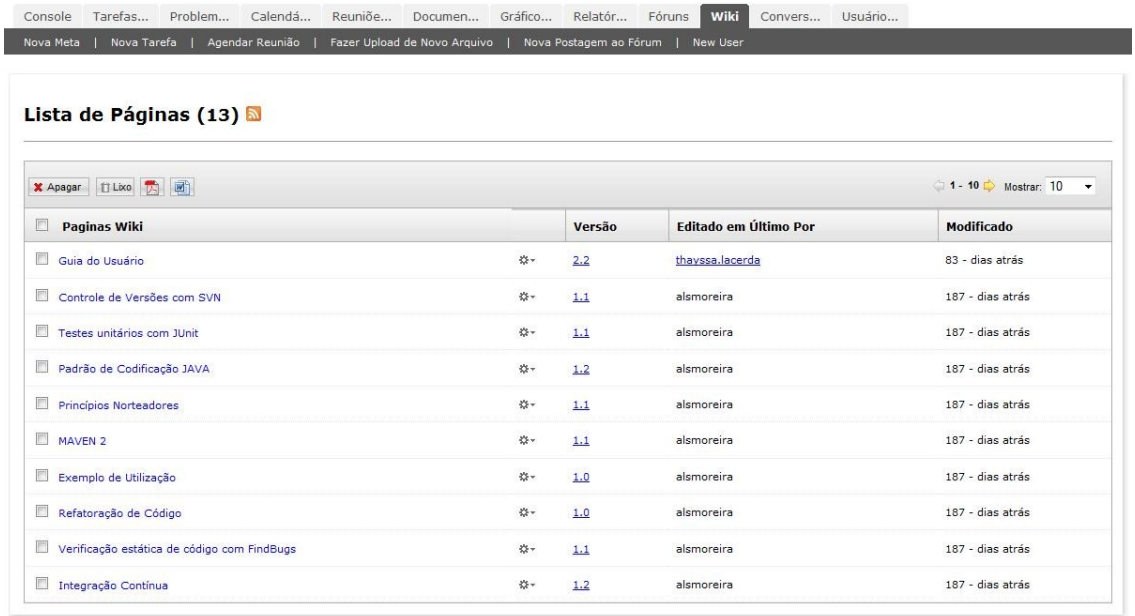

Figura 2 – Guia do Usuário, na ferramenta *Wiki* contida na aplicação *Zoho*.

5. SISTEMAS UTILIZADOS NO PROJETO

A lista de ferramentas abaixo é utilizada para sistematizar a inserção do novo integrante à equipe. Os critérios de escolha dessas ferramentas foram a facilidade de uso, interatividade e, acima de tudo, a familiaridade do usuário, tornando o entendimento mais acessível.

Os sites oficiais, onde estão disponíveis as informações sobre cada ferramenta, estão relatados ao final do artigo como referências na Internet.

5.1 *Wiki*

Tem como principal finalidade a documentação de informações e exibição dos mesmos em ambiente de rede de forma simples e objetiva, o que a diferencia de outras páginas na internet é a fácil edição do texto. Esta flexibilidade auxilia a gestão de conhecimento, visto que a infra-estrutura de projetos e algumas metodologias, bem como as tecnologias sofrem modificações e correções constantes. Assim as informações referentes a modificações seriam rapidamente transferidas aos documentos. O manuseio dessa ferramenta se faz necessário o uso de HTML, CSS e *JavaScript.*

5.2 Zoho

É uma aplicação de escritório *online*, que inclui diversas ferramentas úteis no ambiente de gerencia de projetos, entre elas estão *Zoho Planner*, agenda onde estão presentes horários de reuniões e tarefas executadas pelos membros cadastrados em determinado projeto, *Zoho Wiki*, que possui a mesma finalidade da *Wiki*, porém é limitada à aplicação que pertence, e não precisa do uso de ferramentas como HTML, CSS e *JavaScript, Zoho Chat* e *email*, que facilitam a comunicação dentro do ambiente de desenvolvimento. Possui interface clara e simples é de fácil utilização (ZOHO, 2009).

5.3 *Redmine*

É um *software* livre baseado em ferramenta de gerenciamento de *bugs* e *web*. Entre seus aplicativos estão calendários e diagramas de Gantt, que contém os avanços de diferentes etapas de um projeto, as tarefas de cada membro da equipe, bem como o tempo utilizado para cumpri-la, permitindo a analise do empenho de cada membro no grupo. Tem foco na representação visual do projeto, incluindo *deadlines*. Permite representação de múltiplos projetos, além de suportar diversos bancos de dados.

5.4 *dotProject*

Um sistema de gerência de projetos em software livre de fácil utilização indicado para implementação em ambientes corporativo, por possuir funcionalidades que atendem a muitas necessidades de gerentes de projetos. Por ser uma aplicação web, sua utilização independe de sistema operacional e instalação na máquina do usuário, pois é executado em um servidor.

Possui em suas funcionalidades o diagrama de Gantt, lembretes (*popup)* sobre prazos próximos ao fim, fóruns relacionados a projetos, calendários com visões diferentes: mensal, semanal e diária, entre outras.

6. COMPARATIVO DOS AMBIENTES ANALISADOS

Diante das análises feitas nos ambientes de desenvolvimento, SIEP-Gerencial e DADT, com base nas metodologias utilizadas, foram constatadas as vantagens do uso do Guia do Desenvolvedor *online* em relação à alocação de desenvolvedor experiente para treinamento de um novo integrante. Os novos integrantes adquiriram os conhecimentos básicos e estruturados para garantir uma boa participação no projeto em tempo menor devido ao fácil acesso às informações, o que possibilita o não abandono da produção pelo integrante veterano.

Além disso, o guia se mostra de forma interativa com o usuário e o projeto pela sua flexibilidade em alterações no decorrer do desenvolvimento, fazendo com o mesmo esteja sempre atualizado. Foi observado que a produtividade dos integrantes da equipe de desenvolvimento avançou em média 40% a partir da comparação do tempo de execução do projeto antes e após a aplicação da metodologia.

Modelos como o CMMI (CMMI, 2010) inclui em seu escopo práticas específicas para tratar com inclusão de integrantes em equipe, como estabelecendo mecanismos de gestão organizacional, como a prática do *empowerment*, ou seja, segundo ARAÚJO, 2001, é o fortalecimento do poder decisório dos indivíduos da empresa, ou criação de poder decisório para os indivíduos. Outras práticas são recursos de gráficos e estruturação de equipe. Porém em comparação as práticas do CMMI os projetos que serviram como estudo de caso não tem pessoas com maturidade na área de software suficiente para implementá-las.

A facilidade encontrada no Guia do Usuário é que o conhecimento é organizo de forma adequada e coordenada. Fazendo com que realmente o desenvolvedor precisa aprender. Isso não apenas aumenta a produtividade da equipe como também qualifica de forma sistemática o novo indivíduo. O problema é que um esforço individual é exigido do desenvolvedor que está utilizando o Guia. No núcleo do SIEP-Gerencial, muitas vezes o indivíduo novato que estava utilizando o Guia, ficava disperso e não acompanhava todos os passos do aprendizado.

7. CONCLUSÃO

O presente artigo mostrou como técnicas específicas de integração são de grande importância em uma equipe de desenvolvimento. Equipes que trabalham com essa metodologia devem ter em mente o quanto é importante a rapidez de aprendizado de seus membros. Um sistema bem desenvolvido carrega consigo um grande comprometimento entre a equipe, gerando uma atividade mútua de comunicação e aprendizado.

Observa-se diferentes maneiras de fazer uma integração dentro da equipe como realização de *workshop* do sistema como um todo, explicando as funções de cada um, como também alocar um único desenvolvedor experiente para treinar o novo membro. Tendo em vista que neste ponto a maior dificuldade em integrar é a diminuição no ritmo da produção, pois estaria retirando um membro experiente de suas atividades. Desta forma se faz necessário o guia do desenvolvedor, ou seja, uma disponibilização de procedimentos. Nele, estariam todas as informações decorrentes ao projeto, como por exemplo, ferramentas complementares, as funcionalidades e casos de uso. Neste exemplo não precisaria deslocar um desenvolvedor de suas tarefas, simplesmente estenderia um tempo de treinamento com o guia para que ele pudesse tomar suas atividades no projeto.

Também a comparação do Guia do Desenvolvedor com o Guia do Usuário pode-se observar que ambas as ferramentas apresentadas necessitam de um envolvimento contínuo de um novo integrante. Uma prática que pode ser observada e que pode ser muito útil é que o Gerente do Projeto, durante as reuniões periódicas, deve abrir questionamentos a respeito da ferramenta, desta forma, aprimora-se a mesma e garante uma constante atualização do seu conteúdo.

Por causa dessa técnica durante as reuniões, pode-se remover uma série de erros encontrados nos textos das ferramentas. A técnica de documentação, armazenada na *wiki,* favorece o desenvolvedor não só a leitura como também a escrita de informações pertinentes, devendo um Gerente mais experiente na área de gerencia, sempre rever essas modificações.

É uma abordagem muito interessante, porque todos são responsáveis por melhoria da documentação, contribuindo com o tópico que eles têm mais conhecimento fazendo se sentirem mais confiantes ao documento. E que este tenha um conteúdo mais completo. Uma prática muito útil que pode ser utilizada em um trabalho futuro é o uso de vídeo aulas nas ferramentas de site. Tal trabalho já vem sendo desenvolvido utilizando em parceria alguns recursos de video-treinamento como o Moodle (MOODLE, 2009), que é um ambiente de treinamento virtual muito utilizado em núcleos de educação a distância. Outra forma prática que pode ser desenvolvida em um trabalho futuro é a criação de treinamento em dispositivos móveis, porém o que pode causar problemas é o uso de telas pequenas para leituras. Para sanar tal deficiência pode-se utilizar sistemas de leitores digitais.

REFERÊNCIAS

ARAUJO, L. C. G.. **Organização, sistemas e métodos e as modernas ferramentas de gestão organizacional: arquitetura, benchmarking, empowerment, gestão pela qualidade total, reengenharia.** São Paulo: Atlas, 2001.

CMMI. ***CMMI Overview***. Disponível em: <http://www.sei.cmu.edu/cmmi/>. Acesso em: 15 mai 2010.

CUSUMANO, M.F. ***The software factory: a historical interpretation***, *IEEE Software Magazine*, março, 1989.

DERNIAME, J. C. ***A Comparative Review of Process-Centered Software Engineering Environments***. Annals of Software Engineering 14, p. 311-340. 2002.

Dijkstra, E. W. (Aug 1972). ***The Humble Programmer***. *Communications of the ACM* **15** (10): 859–866. doi:10.1145/355604.361591. (EWD340) PDF, 1972 ACM Turing Award lecture.

EBERT, C. ***Global Software Engineering***. IEEE ReadyNote (e-Book), IEEE Computer Society, Los Alamitos. USA, 2006.

ENAMI, L. N. **Um Modelo de Gerenciamento de Projetos para um Ambiente de Desenvolvimento Distribuído de Software**. Maringá, 2006.

FUGGETTA, Alfonso. **Software Process: A Roadmap**. In Proc. of The Future of Software Engineering, ICSE'2000, Limerick, Ireland.

FUGGETTA, Alfonso; **Support for Software Federations: the PIE Platform.** European Workshop on Software Process Technology, 2000.

HERBSLEB, J. D. ***An Empirical Study of Global Software Development: Distance and Speed***. Proceedings International Software Engineering, IEEE CS Press. Califórnia, 2001.

HERBSLEB, J., AUDRIS, M., THOMAS, A. F., REBECCA, E. G. ***Distance, dependencies, and delay in a global collaboration***, ACM, Computer Supported Cooperative Work, Philadelphia, 2000.

HUGH G. J. AITKEN, ***Scientific Management in Action: Taylorism at Watertown Arsenal****, 1908-1915*, Princeton University Press, Reprint 1985.

LIMA, A. M.; REIS, R. Q. **Uma proposta de ferramenta para execução descentralizada de processos de *software***. Laboratório de Engenharia de Software (LABES). Pará, 2007.

MOITRA, D.; HERBSLEB, J. D. ***Global Software Development***. In proceedings, International Conference on Software Engineering, pp. 81-90. Toronto, Canadá, 2001.

MOREIRA, A. L. S., CAMPELLO, L., FERREIRA, A., CHAVES, T. S. **Processo de Desenvolvimento Distribuído Aplicado ao SIEP-GERENCIAL** – IV Congresso de Pesquisa e Inovação da Rede Norte, Nordeste de Educação Tecnológica, Belém – PA, 2009.

MOODLE, ***Moodle.org: open-source community-based tools for learning,*** Disponível em <http://www.moodle.org/> Acessado em 20 de outubro de 2009

SOARES, A. P. **Processo Ágil de Desenvolvimento de Software para Equipes Separadas Remotamente que Utilizam Ferramentas de Software Livre como Suporte ao Desenvolvimento**. Trabalho de Conclusão de Curso, Centro de Informática - UFPE - Recife, 2009.

STAIR, R. M., REYNOLDS, G. W., **Princípios de sistemas de informação**, Editora Cengage, 8ª edição, 2009.

ZOHO, ***Project Management, Collaboration & Issue Tracking Software***. Disponível em: <http://www.zoho.com/projects/index.html> Acesso em: 10 mar 2010.