METODOLOGIAS ÁGEIS PARA DESENVOLVIMENTO DE SOFTWARE: SELEÇÃO DO MODELO PARA UMA PEQUENA EMPRESA

Maria Inés Castiñeira[1], Carolina Gwozdz Poersch[2], Vera Regiane N. Schuhmacher[3]

Abstract *— Software engineering encompasses different software models: waterfall, spiral, RUP, XP, Crystal, SCRUM, Iconix, etc. The last four, recently proposed, are called agile models. The software development processes are related to the correct use of project management techniques. This paper describes agile models, project management techniques and free software tools to support these activities. A small size software development company was studied. Through an interview and questionnaires, its practices of software development and project management were analyzed. With this information the more suitable agile model and tools for the organization were chosen. A workshop with the model, tools and obtained results were held on the company. This research was developed as a scientific initiation project and the company belongs to a former university student. Thus, in addition to academic research this work is also an example of continuing education for the company members.*

Index Terms —Agile models, project management, SCRUM, Software development process.

1 INTRODUÇÃO

Atualmente a indústria de software e a Tecnologia da Informação (TI) desempenham um papel significativo na economia mundial. Para atingir o pleno desenvolvimento de uma nação esses setores devem ser estimulados a atingir padrões de qualidade. Em 2004 as atividades de software foram eleitas pelo o Governo Federal como opção estratégica da Política Industrial, Tecnológica e de Comércio Exterior a partir da identificação de seu papel crucial para o desenvolvimento das mais diversas atividades [1].

Na área de Engenharia de Software um processo de desenvolvimento de software é um conjunto de atividades, parcialmente ordenadas, que tem a finalidade de obter um produto de software de qualidade. Modelos de processos são metodologias repetíveis e previsíveis propostas com o intuito de melhorar a produtividade e qualidade do produto de software construído [2]. Pressman classifica os modelos em dois grupos: os modelos prescritivos e os modelos ágeis. Entre os modelos prescritivos ou tradicionais podem ser mencionados o modelo cascata, o modelo espiral, a prototipagem e o mais recente Unified Process. Já os modelos ágeis defendem uma constante interação com o cliente, são mais voltados às pessoas e não a processos ou algoritmos, determinam um foco maior na codificação e não na modelagem. São claramente adaptativos, não existe a preocupação de conhecer todos os requisitos funcionais a partir de uma avaliação prévia: novidades são analisadas e incorporadas durante o desenvolvimento. A metodologia ágil é naturalmente iterativa, incremental e acrescenta uma boa participação do usuário no processo. Alguns exemplos são o XP [3], Scrum [4] Crystal e o ICONIX [5].

Como um grande número de projetos de software não atende suas expectativas em termos de funcionalidades, custo, ou cronograma de entrega ainda não existe um modelo de processo perfeito para todas as aplicações e organizações. Modelos ágeis parecem se adaptar melhor para empresas de pequeno ou médio porte [6].

Além das técnicas de engenharia de software técnicas de gerenciamento de projeto também são utilizadas nos processos de desenvolvimento. Em geral, os projetos são componentes críticos da estratégia de negócios de organizações executoras. Eles devem ser planejados, executados e controlados [7]. Desenvolver um software pode virar um caos incontrolável se o projeto não for bem gerenciado.

Desenvolver software é uma atividade complexa, conseguir respeitar os contratos realizados como cliente, nos prazos estipulados, utilizando os recursos (custos) pré-definidos e entregando um produto de software que atinja a qualidade estabelecida ainda continua sendo um desafio para muitas empresas. A adoção de técnicas de engenharia de software são hoje práticas comuns para lidar com essa complexidade. Uma das ações para fortalecer as empresas nacionais de software é promover a qualidade e competitividade, objetivos almejados pelo Programa Brasileiro da Qualidade e Produtividade, PBQP-Software [1].

O uso de técnicas e ferramentas apropriadas no controle dos projetos bem como a utilização do processo de desenvolvimento adequado podem ser fatores críticos

1 Maria Inés Castiñeira, Universidade do Sul de Santa Catarina (UNISUL). Ciência da Computação e Sistemas de Informação. Grande Florianópolis, Santa Catarina, Brasil. Maria.castineira@unisul.br
2 Carolina G. Poersch, Universidade do Sul de Santa Catarina (UNISUL). Estudante de Ciência da Computação. Grande Florianópolis, Santa Catarina, Brasil. tolzinha@gmail.com
3 Vera R.N. Schuhmacher, Universidade do Sul de Santa Catarina (UNISUL). Ciência da Computação e Sistemas de Informação. Grande Florianópolis, Santa Catarina, Brasil. Vera.schuhmacher@unisul.br

quando considerando empresas de desenvolvimento de software de pequeno e médio porte. É objeto de este trabalho promover o desenvolvimento de software que atenda plenamente as necessidades (requisitos) do usuário, dentro dos prazos e orçamentos previstos, favorecendo assim a produção de um produto de qualidade e apoiando dessa forma as empresas desenvolvedoras de software, em particular para aquelas de pequeno e médio porte.

Assim, são objetivos deste trabalho pesquisar técnicas e ferramentas para gerência de projetos, adequadas para empresas de pequeno e médio porte, integração dessas técnicas e ferramentas com os modelos ágeis de desenvolvimento de software e apresentação dessa integração em uma pequena empresa para possível aplicação.

Os objetivos específicos deste trabalho são:

- Pesquisar os modelos ágeis de desenvolvimento de software;
- Pesquisar técnicas e ferramentas de gerência de projetos adequadas para pequenas e médias empresas, preferencialmente ferramentas de software livre;
- Verificar a adequação de técnicas e ferramentas de gerência com algum dos modelos ágeis e elaborar uma proposta com a integração de ambos;
- Contribuir com a disseminação das boas práticas de software, que levam a um produto de qualidade, através da divulgação dos resultados deste trabalho.

Depois da construção do referencial teórico, levantamento de técnicas e ferramentas foi traçado o perfil da empresa participante. A seguir foi escolhido um dos modelos ágeis para construir a proposta integrada com a gerência de projetos. A proposta foi apresentada na empresa de desenvolvimento através de uma palestra, com um posterior debate sobre a aplicação da proposta. Os resultados serão divulgados através do site do grupo de pesquisa (http://inf.unisul.br/~lpu). A seguir são apresentados muito resumidamente alguns conceitos do referencial teórico elaborado, o desenvolvimento da proposta e os resultados obtidos.

2 METODOLOGIAS ÁGEIS E GERÊNCIA DE PROJETOS

Três metodologias consideradas ágeis foram pesquisadas: ICONIX, XP e SCRUM.

O ICONIX é um processo simplificado que unifica conjuntos de métodos de orientação a objetos em uma abordagem completa, com o objetivo de dar cobertura ao ciclo de vida [8]. Geralmente ele é apresentado como uma metodologia prática, intermediária entre a complexidade do RUP (Rational Unified Process) e a simplicidade do XP (Extreme Programming). Segundo Beck [3], a XP é uma metodologia ágil para equipes pequenas e médias que desenvolvem software baseado em requisitos vagos e que se modificam rapidamente. A XP baseia-se em 12 práticas. Algumas dessas práticas causa polêmica à primeira vista e muitas não fazem sentido se forem aplicadas a parte. É a aplicação de todas as regras que sustenta o bom resultado da XP.

A metodologia Scrum é baseada em princípios semelhantes aos da XP como, por exemplo, equipes pequenas, requisitos poucos estáveis ou desconhecidos e iterações curtas para promover viabilidade para o desenvolvimento. A Scrum divide o desenvolvimento em iterações (sprints) de trinta dias. Equipes pequenas, de até dez pessoas, trabalham em cima de funcionalidades definidas no início de cada sprint. A equipe é responsável pelo desenvolvimento desta funcionalidade. Existe também, reuniões de acompanhamento diárias, que são preferencialmente de curta duração (aproximadamente quinze minutos), onde são discutidos pontos como o que foi feito desde a última reunião e o que precisa ser feito até a próxima. As dificuldades encontradas e os fatores de impedimento (bottlenecks) são identificados e resolvidos.

O ciclo de vida da Scrum é baseado em três fases principais, divididas em sub-fases [9]: Pré-planejamento, Desenvolvimento, Pós-planejamento.

2.1 Ferramentas de Software para modelos ágeis

Existem várias ferramentas para modelos ágeis utilizadas para auxiliar os desenvolvedores. Algumas delas são disponibilizadas gratuitamente para download. Alguns exemplos dessas ferramentas são: Scrinch, CVS, XPlanner e VersionOne. Essas ferramentas permitem controlar as versões (configuração), auxiliar nos testes, gerar relatórios, gráficos e definir prioridades, entre outras várias funções relacionadas ao modelo ágil escolhido. Ferramentas de gerenciamento de configuração de software, unidades de teste, e gestão e construção de software são fatores chaves no sucesso de projetos ágeis [10]. Cada uma dessas ferramentas foi testada e analisada para possível uso pela empresa parceira.

2.2 Gerência de projetos

De acordo com Vargas [11], "o gerenciamento de projetos se refere à aplicação de conhecimentos, habilidades, ferramentas e técnicas às atividades do projeto a fim de satisfazer seus requisitos, e é realizado com o uso de processos tais como iniciar, planejar, executar, controlar e encerrar". O tempo, o custo e o escopo são algumas das principais variáveis usadas para execução de alguns projetos. A variável do tempo influencia o prazo até o término do projeto, a de custo, informa o valor monetário incluído no orçamento disponível para o projeto e a de escopo, designa o que deve ser feito até a finalização do projeto. As variáveis estão diretamente ligadas, ou seja, aumentando o escopo, aumenta o tempo e o custo. Diminuindo o tempo, pode significar um aumento no custo e uma diminuição no escopo. E, diminuindo o orçamento, pode significar um aumento de tempo e um escopo reduzido. O PMBOK (Project Management Body of Knowledge) é um conjunto

de práticas em gerência de projetos levantado pelo Project Management Institute (PMI). Essas práticas são bastante utilizadas em áreas como engenharia civil e desenvolvimento de software [7].

No desenvolvimento ágil de software, o projeto é visto como um conjunto de pequenas tarefas, ao invés de um processo completo. Essa abordagem contempla vários métodos ágeis, alguns dos quais já preconizam algumas das práticas de gerenciamento de projetos, embora de uma forma bastante flexíveis, pois permitem ajustes durante a execução dos projetos. Usualmente, a mesma metodologia de desenvolvimento ágil já incorpora atividades e tarefas relacionadas ao gerenciamento. Por exemplo, conforme Kniberg [12] na metodologia SCRUM, existem as reuniões para definir o "Backlog Sprint" (planejamento mensal) e as reuniões diárias (de 15 minutos aproximadamente) para verificar o andamento do projeto. Dessa forma, nas metodologias ágeis a aplicação das práticas sugeridas pelo PMBOK parece ser menos aderente ou adequada.

Existem diversas ferramentas para dar suporte às atividades de gerência de projetos. Essas ferramentas e técnicas podem ser usadas tanto pelo gerente como pelos integrantes da equipe de projeto para organizar seu trabalho e atividades. Entre as diversas ferramentas para gerenciamento de projetos foram pesquisadas somente aquelas de software livre. Alguns exemplos dessas ferramentas são: dotProject, MS Project, OpenProj, Gantt Project entre outros vários softwares disponíveis na internet.

3 SELEÇÃO DO MODELO PARA A EMPRESA

A empresa participante dedica-se ao desenvolvimento de sites e de software em geral, principalmente sistemas para internet. A Floripa 360° (http://www.floripa360.com.br/) é uma empresa com base tecnológica, com o foco no desenvolvimento de soluções computacionais, tanto para internet, quanto para sistemas internos. A empresa conta com sua própria equipe de desenvolvimento, possibilitando assim que seus clientes tenham projetos personalizados. Pode ser classificada como micro empresa, pois possui 5 pessoas ocupadas.

Após contatos telefônicos foi realizada uma entrevista com a apresentação formal do projeto e uma série de questões sobre a forma de desenvolver software da empresa. A continuação foi elaborado um questionário com perguntas abertas e fechadas para identificar o perfil da empresa, o perfil dos produtos de software desenvolvidos e as práticas ou metodologias de desenvolvimento de software, de gerenciamento e de qualidade utilizadas pela empresa. Ele foi aplicado a todos os membros da empresa.

A partir dos resultados da entrevista e dos questionários, percebeu-se que o método mais apropriado para e empresa Floripa 360° é o método SCRUM. Esta escolha foi devida à verificação de que a programação em pares, uma das práticas essenciais da metodologia XP, não seria aplicável na empresa. Por outra parte, a seqüência de atividades definidas pela metodologia ICONIX (casos de uso, protótipos das interfaces, modelo do domínio, diagrama de robustez, etc.) também implicariam em uma mudança muito radical na forma de desenvolvimento da empresa. A utilização da metodologia ICONIX força um peso maior na etapa de modelagem

Dessa forma a escolha recaiu no método SCRUM, pois este permite a adoção conjunta de um modelo de desenvolvimento particular a ser usado em cada SPRINT. Assim, caso a empresa deseje intensificar as práticas do modelo ICONIX, em particular o desenvolvimento dos diagramas de caso de uso, de robustez e seqüência usando a linguagem UML, ela poderia aplicar essa prática dentro de cada Sprint.

Essas informações foram apresentadas à empresa através de um workshop abrindo espaço para o debate e sugestões sobre a possível adequação, uso e resistências das práticas apresentadas. A apresentação contou com a participação dos donos da empresa que participam diretamente do processo de desenvolvimento do software e durou cerca de duas horas, nas quais foram sanadas todas as dúvidas que surgiram no momento da apresentação. O modelo foi apresentado detalhadamente e após a apresentação, os donos da empresa ressaltaram que poderiam vir a utilizar algumas das práticas apresentadas para aprimorar o desenvolvimento de software existente atualmente.

4 CONCLUSÕES E TRABALHOS FUTUROS

O uso do processo de desenvolvimento apropriado e a gerência eficaz de cada projeto -usando as técnicas, modelos e ferramentas adequadas para cada equipe e produto de software- são fatores essenciais para promover a qualidade e produtividade das pequenas e médias empresas desenvolvedoras de software da região. Essas técnicas e metodologias ainda não são utilizadas por todas as empresas de software.

Este trabalho objetivou realizar uma pesquisa sobre modelos ágeis e suas ferramentas, gerência de projetos, ferramentas para gerenciar projetos, pesquisa sobre micro e pequenas empresas, tudo isso a fim de ajudar na produção de um produto de qualidade e apoiando dessa forma as empresas desenvolvedoras de software e promovendo um desenvolvimento que atenda plenamente as necessidades (requisitos) do usuário, dentro dos prazos e orçamentos previstos. Dessa forma, se estará auxiliando as empresas desenvolvedoras a atingir melhor produtividade. As empresas usuárias e população em geral serão atingidas, de forma indireta, através da possibilidade de uso de produtos de software de origem nacional com características de qualidade.

Através do contato e relacionamento com a empresa parceira do projeto percebeu-se que o dia a dia das pequenas empresas desenvolvedoras de software é bastante frenético e a aplicação das teorias, práticas e ferramentas aprendidas no

meio acadêmico nem sempre é possível de forma completa e/ou sistemática. Além disso, a aplicação das práticas de engenharia de software está muitas vezes fortemente relacionada ao uso de ferramentas tecnológicas e essas tecnologias mudam constantemente. O fato dos modelos de desenvolvimento ágil enfatizar mais a interação entre as pessoas envolvidas do que as ferramentas utilizadas é um dos quesitos que os tornam apropriados para as pequenas empresas. Por exemplo, no caso da empresa estudada os donos não apresentaram interesse na indicação e uso de uma ferramenta para gerência de projeto. Na empresa as práticas do gerenciamento são realizadas de forma eficiente sem uma ferramenta específica (apenas utilizando e-mail) devido á quantidade reduzida de funcionários. Esse modus operandi é bem aderente às práticas dos modelos ágeis.

Quanto a adoção do modelo SCRUM a empresa sim mostrou-se interessada, já tinham algum conhecimento sobre esses modelos, mas o tempo foi insuficiente para a aplicação do modelo e verificação de resultados antes da finalização deste trabalho, já que isso requereria um prazo bem maior. De qualquer forma a empresa achou positiva a exposição dos diferentes modelos e as justificavas que apontavam o SCRUM como sendo o modelo mais apropriado para aplicação pela empresa. Como trabalhos futuros diversas ações são possíveis: implantação do modelo SCRUM na empresa, seleção de práticas de engenharia de software (modelagem e desenvolvimento) para serem usadas em cada Spint do SCRUM, estudo dos processos de qualidade no desenvolvimento de software para empresas de pequeno porte, entre outros.

AGRADECIMENTOS

Pelo apoio financeiro, ao CNPq (Carolina G. Poersch: Programa PIBIC) e à Coordenação de pesquisa da Unisul.

REFERÊNCIAS BIBLIOGRÁFICAS

[1] BRASIL. Agência Brasileira de Desenvolvimento Industrial. *Política Industrial, Tecnológica e de Comércio Exterior (PITCE).* Disponível eletronicamente em http://www.abdi.com.br/abdi_redesign /publicacao/engine.wsp?tmp.area=332. Dados acessados em Março de 2007a.

[2] PRESSMAN, R. S. *Engenharia de Software*. 6. ed. São Paulo: Makron Books, 2006.

[3] BECK, K. *Extreme Programming Explained*: Embrace Change. Addison-Wesley, Boston, USA, 1999.

[4] SCHWABER, K.; BEEDLE, M. *Agile Software Development with SCRUM.* Prentice Hall, Series in Agile Software Development, 2001.

[5] ROSENBERG, D; STEPHENS, M, COLLINS-COPE, M. *Agile development with ICONIX process: People, process, and pragmatism*. Apress, USA, 2005.

[6] COHEN, D., LINDVAL, M., & COSTA, P. *An introduction to agile methods.* In Advances in Computers. Elsevier Science, New York, (p. 1-66), 2004.

[7] CORDEIRO Martins, J.C. *Gestão de projetos de Desenvolvimento de Software: PMI- UML.* Ed. Brasport, RJ, 2002

[8] ROSENBERG, D; SCOTT, K. *Use Case Driven Object Modeling with UML: A Practical approach.* Massachusetts: Addison-Wesley, 1999.

[9] SOARES, M. S. *Metodologias Ágeis Extreme Programming e Scrum para o Desenvolvimento de Software*. 2002. Material didático. UFRGS. Disponível Eletronicamente em: <http://www.inf.ufrgs.br/~zirbes/MaterialAula/>. Acessado em outubro de 2008.

[10] BARNETT, Liz; SCHWABER, Carey. *Agile Development Teams Need Tools, Too.* 2004. Disponível eletronicamente em: http://www.forrester.com/Research/Document/Excerpt/0,7211,34977, 00.html . Acesso em Janeiro de 2009.

[11] VARGAS, Ricardo Viana. *Gerenciamento de projetos: estabelecendo diferenciais competitivos.* Rio de Janeiro: Brasport, 2000.

[12] KNIBERG, Henrik, *SCRUM e XP direto das fronteiras: Como nós fazemos SCRUM.* Editora: C4Media, Publisher of InfoQ.com. Série InfoQ de livros sobre Desenvolvimento de Software Corporativo. 2007. Versão eletrônica disponível em: http://infoq.com/br/minibooks/scrum-xp-from-the-trenches. Acesso em Dezembro de 2008