

METODOLOGIAS DE DESENVOLVIMENTO DE SOFTWARE NO PROJETO MES DA BELGO MINEIRA¹

*Frederico Medina Vargas²
Douglas Fernando da Silva³
Lourival José Ribeiro Castro⁴
Renata Ferreira Rodriguez⁵*

Resumo

Já há algum tempo o software vem sendo utilizado para acompanhar processos e agilizar tomada de decisão. A possibilidade de desenvolvimento de softwares específicos para cada necessidade, aumentou a demanda por esse serviço e profissionalizou o desenvolvimento, com o surgimento de várias metodologias. Este trabalho apresentará uma discussão sobre a metodologia utilizada no desenvolvimento do projeto MES da Companhia Siderúrgica Belgo Mineira e as modificações feitas no modelo original para melhor adaptação ao projeto. A seleção de uma metodologia de desenvolvimento de software não foi tarefa simples devido a uma série de características do projeto, sendo a principal delas seu ousado cronograma. A solução adotada foi montar uma metodologia baseada nas existentes, a fim de tornar o processo o mais dinâmico possível, sem comprometer a qualidade do produto final. Um líder técnico foi responsável pelas definições técnicas estratégicas e pela intermediação entre a equipe de especificação e a de desenvolvimento. Desenvolvimento e implantação foram divididos em módulos, possibilitando melhor divisão de tarefas e menor impacto no ambiente como um todo. A utilização de uma metodologia de desenvolvimento de software adequada tem se mostrado fundamental no sucesso dos projetos. Em um projeto de grande porte, como o desenvolvimento do sistema MES da Belgo, esse ponto se torna ainda mais crítico. Como resultado, chegou-se há um sistema de excelente qualidade técnica, com grande nível de reaproveitamento de código e entregue dentro dos prazos negociados.

Palavras-chave: Metodologias de desenvolvimento de software; MES; Qualidade de software.

¹ IX SEMINÁRIO DE AUTOMAÇÃO DE PROCESSOS, 5 a 7 de outubro de 2005 – Curitiba – PR

² Líder de Projetos, Chemtech

³ Engenheiro de Aplicações Chemtech

⁴ Analista de Sistemas, BMS

⁵ Analista de Sistemas, BMS

A EVOLUÇÃO DA PRODUÇÃO DE SOFTWARE

Assim como a tecnologia, o processo de desenvolvimento de software sofreu e vem sofrendo muitas mudanças nos últimos anos. As metodologias tradicionais, também conhecidas como pesadas ou orientadas a documentação, vem sendo substituídas pelas chamadas metodologias ágeis de desenvolvimento de software.

Essa mudança de cenário está sendo promovida por uma necessidade de adaptação do processo de desenvolvimento de software frente às novas tecnologias disponíveis no mercado e também às expectativas dos clientes que, cada vez mais, exigem produtos de alta qualidade, desenvolvidos com baixo custo e em prazos cada vez menores.

É fácil compreender a baixa aderência das metodologias tradicionais à nova realidade do mercado, já que muitas delas foram concebidas em um contexto muito diferente do atual, onde os sistemas eram principalmente baseados em mainframe. Naquela época, o custo de fazer alterações e correções era muito alto, uma vez que o acesso aos computadores era limitado e não existiam modernas ferramentas de apoio ao desenvolvimento de software, como depuradores e analisadores de código. Essa realidade fazia com que fosse imprescindível planejar e documentar todo o software antes da fase de implantação.⁽¹⁾

O dinamismo dos sistemas atuais fez com que as metodologias se tornassem mais dinâmicas e capazes de tornar o processo de desenvolvimento de software mais flexível e aberto a mudanças.

Mesmo com a evolução dos computadores, das ferramentas e dos conhecimentos de engenharia de software nos últimos anos, a produção de software confiável, de boa qualidade e entregue dentro dos prazos e custos estipulados ainda é muito difícil.

METODOLOGIAS ÁGEIS X METODOLOGIAS TRADICIONAIS

O grande diferencial das metodologias ágeis está no fato delas serem mais bem preparadas para atenderem a mudanças nos requisitos. Ao contrário das metodologias tradicionais, as ágeis usam técnicas adaptativas ao invés de preditivas para lidar com as mudanças que são frequentes na maioria dos projetos. Sendo assim, pode-se dizer que não tentam prever e planejar como tudo será realizado do início ao fim de todo o projeto para só depois começar a construção, como nas metodologias tradicionais.⁽²⁾

Porém, as técnicas adaptativas necessitam de realimentação do processo de desenvolvimento para poderem tratar essas mudanças de requisitos. Essa realimentação é obtida através de desenvolvimento iterativo em que o resultado de uma iteração é usado como entrada para a próxima iteração. Cada iteração pode ser vista como se fosse um micro projeto nos moldes das metodologias tradicionais.

Com o uso de iterações, técnicas preditivas podem ser utilizadas para iteração atual quando sua eficiência é bem maior, dado que o período de tempo é menor. O resultado da interação é visto rapidamente e qualquer mudança pode ser feita com mais flexibilidade. Os usuários finais também podem ver o resultado das iterações e

fazerem críticas e sugestões. Ou seja, o usuário final vê os resultados do projeto mais rápido e mais vezes, não só no final do projeto quando é mais difícil fazer mudanças. Nesses momentos o usuário final pode ajudar a identificar problemas e sugerir mudanças, quando ações corretivas podem ser adotadas em menos tempo, com menos impactos e custos. As Figuras 1 e 2 apresentam respectivamente a evolução do custo da mudança em um projeto que utiliza uma metodologia tradicional e a evolução do custo da mudança em um projeto que utiliza uma metodologia ágil. Pelos gráficos é fácil notar que o projeto desenvolvido utilizando uma metodologia ágil é mais adaptativo.⁽²⁾

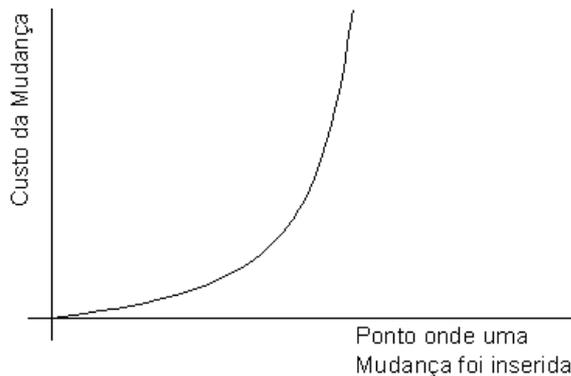


Figura 1. Custo de inserção de uma mudança no projeto conduzido por uma metodologia tradicional.

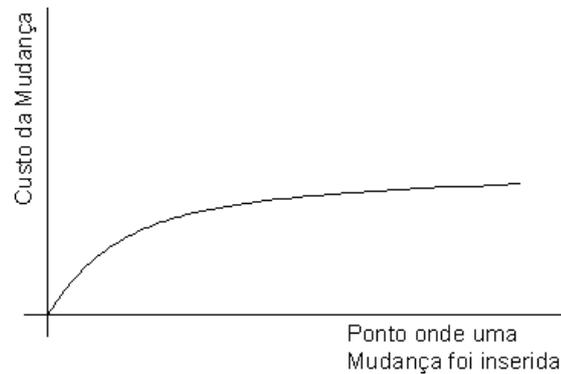


Figura 2. Custo de inserção de uma mudança no projeto conduzido por uma metodologia ágil.

Mas é importante deixar claro que, assim como qualquer tipo de projeto, um projeto de desenvolvimento de software também está sujeito há problemas. Os mais comuns são o não cumprimento de prazos e orçamentos, qualidade do produto inferior às expectativas do cliente (tanto do ponto de vista funcional, quanto do ponto de vista de qualidade), escolha tecnológica não adequada à realidade do sistema, etc. Algumas dessas situações podem levar o projeto a ser até cancelado antes da sua conclusão.

Para reduzir a incidência dos problemas acima relacionados, alguns pontos considerados críticos no processo de desenvolvimento de software precisam ser observados, tais como:

- **Gestão de requisitos:** as mais importantes decisões do projeto são tomadas na fase inicial de levantamento de requisitos. Um levantamento cuidadoso dos requisitos de um software pode ser capaz de reduzir drasticamente os riscos do projeto;
- **Gestão de mudanças:** as expectativas com relação ao produto que está sendo construído mudam à medida que o projeto avança. Uma melhor compreensão do problema e do sistema estimula o surgimento de novos requisitos. Cuidar para que as novidades não comprometam o projeto é de suma importância;
- **Gestão de riscos:** ações pró-ativas para tratar incertezas, avaliando riscos continuamente e usando-os para influenciar a tomada de decisões durante todo o projeto;

- Gestão da qualidade: seja qual for a metodologia, ferramenta ou plataforma utilizadas, a qualidade de software precisa ser garantida. É ela que vai permitir que o produto gerado obtenha sucesso;

Com isso, é fácil perceber a importância da metodologia de desenvolvimento de escolhida para o sucesso de um projeto. O próximo tópico tratará da metodologia de desenvolvimento de software aplicada no desenvolvimento do projeto MES⁽³⁾ da Companhia Belgo Mineira.⁽⁴⁾

A METODOLOGIA DE DESENVOLVIMENTO DO MES DA BELGO MINEIRA

O projeto MES⁽³⁾ da Companhia Siderúrgica Belgo Mineira⁽⁴⁾ surgiu como um grande desafio: tratava-se de um projeto com requisitos extremamente complexos e com um prazo de desenvolvimento bastante desafiador.

Para cumprir essa missão, a gerência do projeto observou que a simples adoção de uma metodologia de desenvolvimento de software não seria capaz de suprir as necessidades do projeto, mas que seria necessário lançar mão de um conjunto de práticas de diferentes metodologias já existentes, a fim de gerar um processo capaz de atender ao projeto.

Nos tópicos a seguir serão apresentadas, de maneira detalhada, algumas práticas utilizadas pelas empresas participantes durante o desenvolvimento desse projeto.

O Organograma do Projeto

Para facilitar o gerenciamento do projeto, o sistema foi dividido em módulos que pudessem ser desenvolvidos paralelamente e independentemente. A Figura 3 ilustra o organograma do projeto.

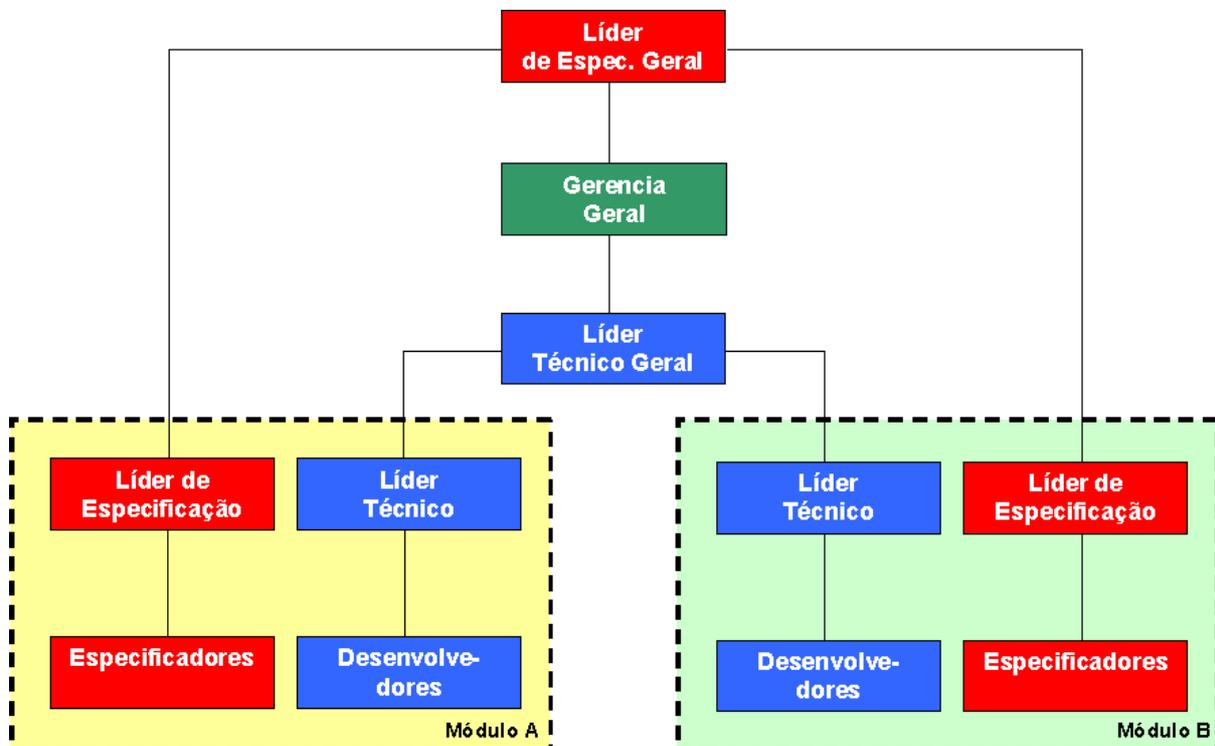


Figura 3. Organograma do projeto MES da Belgo Mineira.

Cada módulo possuía em seu corpo técnico, especificadores (pessoas com conhecimento de processo) e desenvolvedores (programadores e especialistas em desenvolvimento de software). Para coordenar os trabalhos, para cada módulo foram nomeados um líder de especificação e um líder técnico.

O líder técnico do módulo tinha como funções:

- Acompanhar o cronograma de desenvolvimento;
- Zelar pela qualidade do código gerado, padronização de soluções e reaproveitamento do código;
- Revisar cada especificação técnica antes de sua implementação, orientando o programador responsável;
- Acompanhar a correção de erros, preocupando-se com o índice de retrabalho gerado pelos erros de especificação e de codificação;
- Garantir a constante comunicação entre especificação e desenvolvimento;
- Decidir sobre questões técnicas, juntamente com o líder técnico do projeto, sobre os problemas técnicos em geral.

O líder de especificação do módulo tinha como funções:

- Acompanhar o cronograma de especificação e testes;
- Zelar pela qualidade e padronização dos documentos gerados, através da revisão de todos os artefatos antes da entrega ao desenvolvimento;
- Garantir que todos os documentos tenham sido devidamente validados pelos usuários responsáveis antes da entrega ao desenvolvimento;
- Reportar ao líder técnico do módulo sobre eventuais problemas de qualidade no desenvolvimento ou de modificação de especificação;
- Zelar pela padronização das soluções entre as usinas;
- Acompanhar, junto ao líder técnico, a solução dos eventuais problemas detectados durante os testes.

Ainda conforme a Figura 3, no organograma do projeto existiam mais outras duas funções importantes: a do líder técnico do projeto e a do líder de especificação do projeto. Essas pessoas eram responsáveis pelo acompanhamento do desenvolvimento e da especificação do projeto como um todo, sempre procurando garantir a qualidade e o cumprimento do cronograma do projeto. O gerente geral era o responsável por acompanhar todo o projeto com as informações passadas pelos líderes do projeto (de especificação e técnico).

As decisões técnicas estratégicas do projeto eram tomadas pelo comitê técnico (Figura 4), composto pelo líder técnico do projeto, pelo líder técnico de cada módulo e pelo responsável pela infra-estrutura da Belgo.

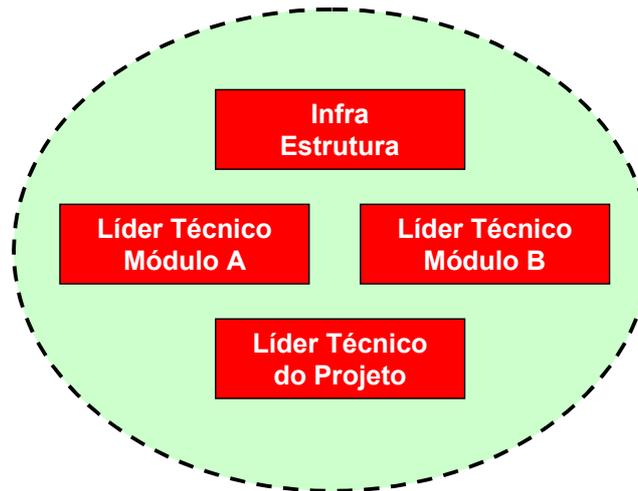


Figura 4. Comitê técnico.

Uma equipe que participou em paralelo de todo o processo é o chamado comitê executivo. Esse comitê, formado pela diretoria de cada uma das empresas envolvidas no processo e era o responsável pela aprovação de grandes mudanças no projeto, tais como: cronograma, escopo do projeto e negociações diversas. A Figura 5 ilustra o organograma do comitê executivo.

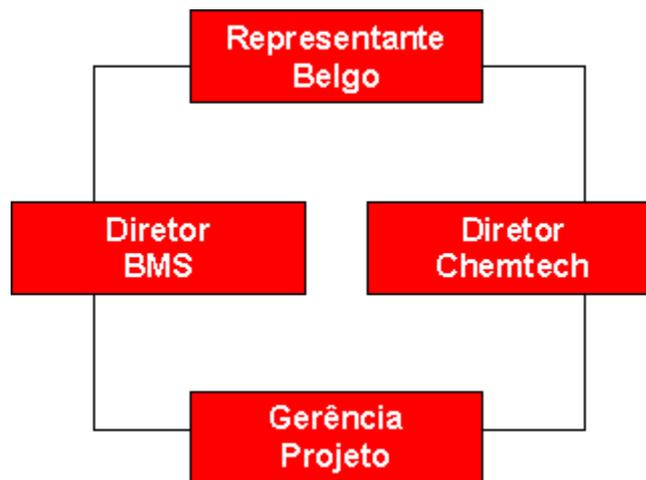


Figura 5. Comitê executivo

O Cronograma do Projeto

Como já foi dito anteriormente, o cronograma do projeto era bastante ousado. Garantir seu cumprimento não seria fácil. Portanto, muito mais do que um cronograma, seria necessário traçar uma eficiente estratégia para que o projeto pudesse terminar no prazo determinado.

Sendo assim, além do desenvolvimento em módulos, o start-up foi planejado para ser realizado separadamente para cada módulo. Com isso, todo o processo de testes, treinamento dos usuários e preparação do ambiente ficou diluído no tempo, o que reduziu a demanda de recursos de infra-estrutura e a mobilização de recursos das usinas, ou seja, o impacto no ambiente como um todo foi reduzido e o

gerenciamento desta etapa, que de uma forma geral é complicado, ficou bastante simplificado.

O Dia-a-Dia do Projeto

No dia-a-dia do projeto várias rotinas foram estabelecidas para melhor controlar a qualidade e o andamento do projeto. Seguem abaixo as mais relevantes:⁽⁵⁾

- Revisão de código: o controle de qualidade do código gerado era feito através da revisão de código. Semanalmente os programadores verificavam o código de um outro colega utilizando como padrão de verificação um check-list. O resultado desta revisão era repassado ao responsável pelo código que imediatamente corrigia os problemas identificados;
- Reuniões técnicas por módulo: os líderes de cada módulo faziam semanalmente reuniões com os seus programadores a fim de identificar gargalos e discutir problemas comuns, assim como obter sugestões. Quando não resolvidos no âmbito dessa reunião esses problemas eram discutidos com o líder técnico do projeto e em alguns casos, com o comitê técnico;
- Construções diárias: todos os dias a solução era construída a partir do conteúdo do controle de versões e colocada num servidor de construção. Com essa versão diária do sistema foi possível observar a evolução contínua do sistema, além de permitir a identificação de possíveis problemas mais rapidamente e garantir a integridade do controle de versões;
- Inspeção de código automática: foram utilizadas diversas ferramentas de análise de código para busca de bugs, más práticas de programação, fuga de padrões, etc. Essas ferramentas permitiram encontrar bugs que são difíceis de se encontrar por revisões de código manual;
- Divulgação de técnicas e práticas: todo membro da equipe quando fazia algum componente que poderia ser utilizado pelos demais notificava a criação do mesmo para os demais membros a fim de aumentar a reutilização de código pela equipe. Além de componentes, houve a divulgação constante de boas práticas e técnicas de membros mais seniores para os demais;
- Testes funcionais e unitários automáticos: as partes do sistema que eram mais críticas tinham testes funcionais automáticos capazes de simular toda a rotina de utilização do sistema. Além de testes funcionais, componentes mais críticos tinham testes unitários que eram também executados automaticamente. Todos os testes automáticos eram auxiliados por ferramentas;

CONSIDERAÇÕES FINAIS

Muitas das estratégias apresentadas nesse documento surgiram no decorrer do projeto como alternativa aos processos sugeridos pelas metodologias existentes na literatura que não poderiam ser implementadas na sua essência devido ao escasso tempo disponível. A principal consequência disso foi o comprometimento de parte do cronograma do projeto.

REFERÊNCIAS BIBLIOGRÁFICAS

- 1 SOARES, M. S. Comparação entre metodologias ágeis e tradicionais para o desenvolvimento de software. Disponível em <http://www.dcc.ufla.br/infocomp/artigos/v3.2/art02.pdf>. Acessado em 05 de Junho de 2005.
- 2 FOWLER, M. The new methodology. Disponível em <http://www.martinfowler.com/articles/newMethodology.html>. Acessado em 15 de junho de 2005.
- 3 ALVES, C. H. N., Almeida, P. I. F. Integração do chão-fábrica com o sistema empresarial, uma realidade que veio para ficar. Revista Intech Brasil nº 71, 2005.
- 4 CORREA, L. C. M., SOUSA, C. E., SILVA, F. W. M. MES Belgo Mineira laminação: Sistema integrado de informações industriais da Belgo. **Anais do IX Seminário de Automação de Processos**, Associação Brasileira de Metalurgia e Materiais, out. 2005
- 5 SOARES, M. S. Metodologias ágeis Extreme Programming e Scrum para o Desenvolvimento de Software. Disponível em <http://www.presidentekennedy.br/resi/edicao04/artigo06.pdf>. Acessado em 05 de Junho de 2005.

SOFTWARE DEVELOPMENT METHODOLOGY IN THE MES PROJECT OF BELGO MINEIRA¹

*Frederico Medina Vargas²
Douglas Fernando da Silva³
Lourival José Ribeiro Castro⁴
Renata Ferreira Rodriguez⁵*

Abstract

Software is used to manage processes and speed decisions. The possibility to develop specific softwares for each problem has increased the demand for this service, professionalizing the development with the creation of many methodologies for software development. This work will discuss the software development methodology used in the MES project for the Companhia Siderúrgica Belgo Mineira and the necessary changes to the original model for a better adapt to the project needs. The selection of a software development methodology was not a simple task because of some project's characteristics like it's hard scheduling. The better solution found was to build a new methodology based on the existents, to make the process more dynamic, keeping the product quality levels. A technical leader was responsible by the technical strategies and to intermedate the contact between the specification and the development team. Development and startup were divided into modules, to improve de task's division and reduce the impact in the environment. The usage of an adequate methodology is essential for the success of all software projects. In a big project, likethe MES system for Belgo, this point gets more critical. The result was a high quality system with a good level of code reuse, finished into the negotiated schedule.

Key-words: Software development methodologies; MES; Software quality

¹ XI SEMINÁRIO DE AUTOMAÇÃO DE PROCESSOS, October, 5 to 7, 2005 – Curitiba – PR – Brazil

² *Líder de Projetos, Chemtech*

³ *Engenheiro de Aplicações, Chemtech*

⁴ *Analista de Sistemas, BMS*

⁵ *Analista de Sistemas, BMS*