

AULA 06 – Tópicos Avançados

Introdução:

Em nossa sexta aula, falaremos sobre alguns assuntos relacionados aos rumos da engenharia de software. Falar em tópicos avançados é algo bastante amplo, mas iremos pontuar três tópicos que se referem à melhorias no processo de desenvolvimento com o objetivo de desenvolvermos um software com qualidade e aderente às necessidades dos stakeholders.

Contextualizando:

Como verificamos em todas nossas aulas, o grande foco é produzirmos um software com qualidade e que realmente seja útil. A princípio pode parecer algo tão simples, mas quando falamos de software identificamos que sempre partimos de desejos, necessidades e vontades humanas que serão transformados em um “software” e o mesmo deverá devolver resultados com um bom funcionamento e uma excelente interatividade. Então, estamos diante de um trabalho bastante desafiador, e é nesta linha de raciocínio que iremos explorar melhoria do processo, desenvolvimento ágil e novas direções da tecnologia como ponto de partida deste encontro.

MELHORIA DO PROCESSO

Melhoria de processo de software implica em elementos de um processo eficaz e alinhada à cultura organizacional, pois necessita de apoio estratégico para definição e aplicações. Toda melhoria de processo deve presumir a obtenção de retorno do investimento, logo, ela envolve maturidade do processo de software e deve fornecer indicadores de qualidade para mensuração. Não há um modelo universal, porém ela sempre focará em processo e produto.

Um dos modelos mais difundidos no mercado é o CMM (Capability Maturity Model), que é estruturado em 5 níveis:

5 – Otimizado

4 – Controlado

- 3 – Definido
- 2 – Reproduzível e
- 1 – Inicial.

A figura 1 descreve qual é o foco de cada nível do CMM.

Nível	Foco	Áreas de Processo
Otimizante	<i>Melhoria contínua do processo</i>	Inovação organizacional e distribuição (deployment) Análise causal e resolução
Controlado quantitativamente	<i>Gerenciamento quantitativo</i>	Desempenho de processo organizacional Gerenciamento quantitativo de projeto
Definido	<i>Padronização de processo</i>	Desenvolvimento de requisitos Solução técnica Integração de produto Verificação Validação Foco no processo organizacional Definição de processo organizacional Treinamento organizacional Gerenciamento de projeto integrado Gerenciamento de fornecimento integrado Gerenciamento de risco Análise de decisão e resolução Ambiente organizacional para integração Equipe integrada
Repetível	<i>Gerenciamento básico de projeto</i>	Gerenciamento de requisitos Planejamento de projeto Monitoração e controle de projeto Gerenciamento de acordo com fornecedor Medição e análise Garantia de qualidade de processo e produto Gerenciamento de configuração
Executado		

Figura 1 – Descritivo dos níveis do CMM

A implantação do CMM é algo extremamente dispendioso e como a maioria das empresas de software possuem recursos financeiros escassos por serem pequenas, utilizam outro modelo de melhoria organizacional como roteiro para iniciar e planejar ações de melhoria, denominado IDEAL. O IDEAL foi desenvolvido pelo Software Engineering Institute.

Independente da abordagem adotada, o importante em ambas é considerar questões na comunicação com o cliente, métodos para representar os requisitos e definição de estrutura de gerenciamento de projeto. E o modelo adotado deve ser

consistente, sofisticado, aceito e comprometido com a organização. Uma questão importantíssima encontra-se no treinamento de toda a equipe envolvida. Alguns fatores que garantirão o sucesso são: comprometimento da gerência, envolvimento do pessoal, integração e entendimento do processo, estratégia personalizada e sólido gerenciamento de projeto. Outras formas de melhoria:

- **SPICE** - iniciativa internacional para suportar a avaliação de processo da ISO e padrões de processo de ciclo de vida [SPI99]
- **ISO/IEC 15504** para Avaliação de Processo (Software) [ISO08]
- **Bootstrap** - uma estrutura SPI para organizações de pequeno e médio porte em conformidade com SPICE [Boo06]
- **PSP e TSP** - estruturas SPI individuais e específicas de equipe ([Hum97], [Hum00]) que se concentram no processo em detalhes, uma abordagem mais rigorosa do desenvolvimento de software combinada com a medição
- **TickIT** - um método de auditoria [Tic05] que avalie se uma organização está em conformidade com a Norma ISO 9001:2000

DESENVOLVIMENTO ÁGIL

Os principais objetivos do desenvolvimento ágil encontram-se na aceleração do processo de desenvolvimento de software, na melhoria contínua do processo, no aumento da comunicação e interação da equipe, na definição de metas, na elaboração de alternativas para falhas, em respostas rápidas às mudanças e no aumento da produtividade. A figura 2 demonstra a participação no mercado de TI dos principais métodos ágeis.

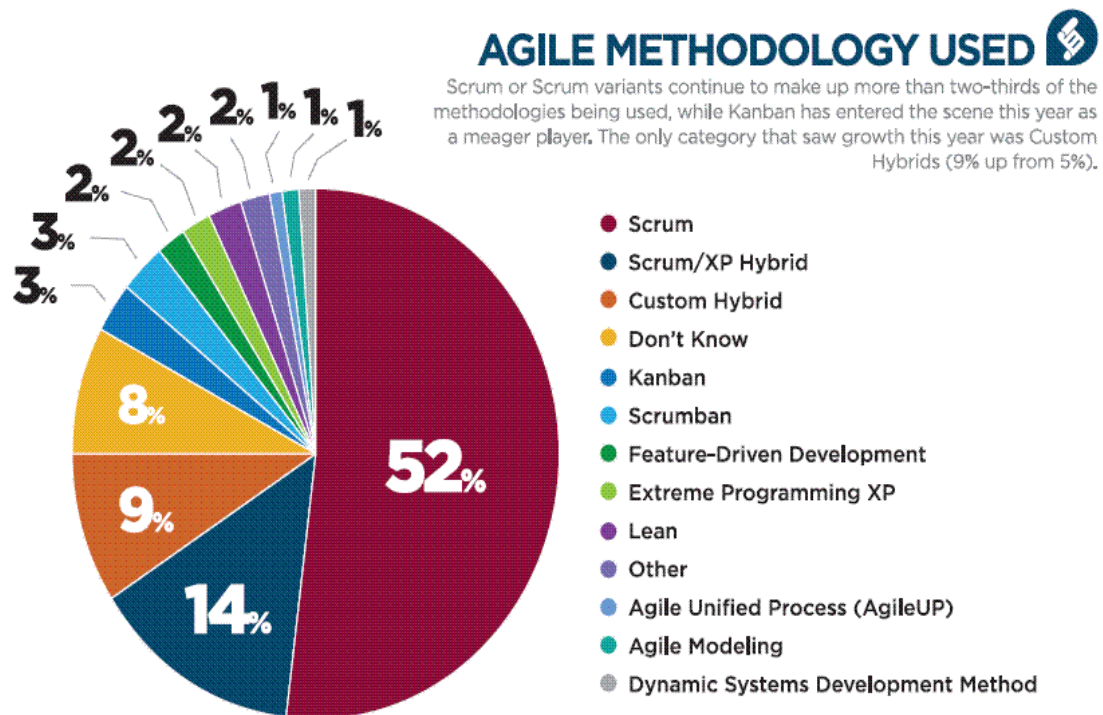


Figura 2 – Métodos ágeis mais utilizados no mercado

Há um manifesto público sobre os doze princípios a serem considerados num método ágil:

- 1 - Maior prioridade é satisfazer ao cliente com entregas contínuas e adiantadas de software com valor agregado.
- 2 – Mudanças de requisitos são bem-vindas sempre. Processos ágeis tiram vantagem das mudanças visando competitividade para o cliente.
- 3 - Entregas frequentes de software funcionando em poucas semanas ou meses.
- 4 – Time: stakeholders e desenvolvedores trabalhando juntos diariamente.
- 5 – Construção de projetos em torno de pessoas motivadas. Ambiente e suporte necessários e confiar a eles o trabalho.
- 6 – Transmissão de informação para equipe e entre equipe. Conversa frente a frente.
- 7 - Medida primária de progresso: software funcional.
- 8 – Desenvolvimento sustentável. Ritmo constante sempre.
- 9 – Bom design e excelência técnica aumentam a agilidade.
- 10 – Simplicidade – a arte de maximizar quantidade de trabalho não realizado.
- 11 - Melhores arquiteturas, requisitos e design emergem de times auto-organizáveis.

12 – Reflexão sobre como tornar mais eficaz, o refinamento e ajuste de comportamento em intervalos regulares.

DIREÇÕES DA TECNOLOGIA

A tecnologia da informação possui um futuro bastante amplo complexo para construção de software. Entre tais aspectos encontram-se:

- 1 – Aplicações ligadas ao contexto
- 2 – Computação Invasiva
- 3 – Software aberto
- 4 – Segurança, confiança e privacidade
- 5 – Clientes com menos requisitos
- 6 – Utilização da filosofia ágil
- 7 – Mix de talentos na equipe de desenvolvimento (antes caro, hoje rápido)
- 8 – Tecnologias como grid computing, computação aberta, microcomércio, máquinas cognitivas, monitores OLED, RFIDs e Web 2.0.

Diante deste novo horizonte faz-se necessário a especialização e conhecimento de algumas linguagens de programação, as quais se adaptam a contextos, ambientes, plataformas e tecnologias diferentes. O futuro da área de TI constitui-se num panorama multidisciplinar, focado no ser humano e ajustado a várias plataformas e ambientes.

Pesquisa

Desenvolva uma pesquisa sobre um dos itens listados como desafios da tecnologia da informação.

Trocando Ideias

Metodologias ágeis são cada vez mais utilizadas, em especial em projetos de pequeno e médio porte. A maioria delas não aborda o uso de diagramas e modelos tais como os utilizados pela UML. Discuta sobre a utilização de modelos e diagramas para modelagem de sistemas em confronto com os métodos ágeis.

Síntese

Nessa aula traçamos algumas questões relacionadas ao futuro do desenvolvimento de software. As questões tanto de tecnologia quanto de necessidades de construção de software são extensas e abordam uma complexidade que cresce em escala. A preparação em vários ambientes e várias linguagens de programação traz benefícios aos profissionais de desenvolvimento de software.

Compartilhando

Busque mais informações sobre CMM e CMMI em relação às questões da maturidade do processo de desenvolvimento de software.

Autoavaliação

- X?