

Resumo da apresentação de Métodos Ágeis

Alunos:

João Machini de Miranda
Lucianna Thomaz Almeida
Pedro Borges Morhy Leal
Thiago Rodrigues Colucci

Os métodos ágeis são uma maneira de pensar em desenvolvimento de software. Ela foi idealizada por experientes desenvolvedores de software que raramente falhavam na entrega e satisfação dos clientes. Em 2001, data em que eles se reuniram, tais desenvolvedores definiram o que eles faziam de diferente dos demais. Assim surgiu o Manifesto Ágil.

O Manifesto é simplesmente uma exposição de valores comuns (ou que deveriam ser comuns) a todos os projetos de software:

“Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.”

(<http://agilemanifesto.org>)

Tendo estes valores como base para o desenvolvimento de software, o projeto se torna mais flexível por não ter um grande design logo no início nem seguir um plano a risca.

Breve explicação de cada item do manifesto:

- **Indivíduos e interações** - significa que a interação entre clientes e desenvolvedores é mais importante e útil do que processos engessados e ferramentas que “auxiliam” o desenvolvimento. É mais prático e claro falar com o cliente diretamente do que usar um sistema, o qual o cliente pode não saber usar e, se souber, precisa escrever o que precisa com suas palavras, e isso pode resultar em textos ambíguos e difíceis de entender;
- **Software funcionando** - apesar da documentação ser importante para explicar o projeto, se o software não possuir as funcionalidades desejadas pelo cliente, o projeto não tem valor. Logo não podemos nos prender em criar documentos e deixar o software ruim ou sem uma versão parcial utilizável para o usuário final;
- **Colaboração com o cliente** - ficar restrito à contratos é simplesmente evitar mudanças e não aceitar o fato de que o cliente não sabe o que ele quer logo

no início. Dessa forma, o software produzido acaba não atingindo a satisfação dos usuários finais, pois o que eles realmente precisam, muitas vezes, não está claro para ambas as partes no contrato. Por isso, a colaboração com o cliente é mais valorizada que contratos de negócios;

- **Responder às mudanças** - dado que a interação entre os programadores e o cliente durante o desenvolvimento do projeto será grande, os requisitos poderão mudar bastante com o tempo. Funcionalidades extremamente necessárias no início do projeto, tornam-se obsoletas e desnecessárias ao final do mesmo. Assim como funcionalidades nem imaginadas no início, no final tornam-se a alma do negócio. Essas mudanças não podem ser evitadas, elas devem ser valorizadas e incorporadas ao projeto para que o software satisfaça o usuário final.

Uma grande característica dessa metodologia é o contrato de escopo variado. Esse tipo de contrato permite que ambos os lados parem o desenvolvimento a qualquer momento. Assim, o cliente tem o poder de dizer que o software está pronto e não precisa de novas funcionalidade (mesmo que não cumpra o “combinado” no início) e a equipe de desenvolvimento pode parar de trabalhar se o cliente não estiver interagindo de forma “produtiva”. Além do desenvolvimento poder ser finalizado a qualquer momento, o escopo variável também permite que as funcionalidades sejam adaptadas de acordo com a necessidade.