

Programação eXtrema (XP) x Engenharia de Software

Alex Cardoso¹, Aline Teixeira¹, Dirceu Pereira Tiegs¹, Fernando Silvano Gonçalves¹, Leonardo da Silva¹, Adson Fernandes Caldas², Rafael Charnovski³.

¹ Acadêmico do curso de Ciência da Computação – Departamento de Ciência da Computação – Universidade do Extremo Sul Catarinense (UNESC) – Criciúma / SC

² Professor do curso de Ciência da Computação – Departamento de Ciência da Computação - Universidade do Extremo Sul Catarinense (UNESC) – Criciúma / SC

³ Professor do curso de Ciência da Computação (até 04/2006) – Departamento de Ciência da Computação - Universidade do Extremo Sul Catarinense (UNESC) – Criciúma / SC

alexkardozo@hotmail.com, aline.teixeira@eliane.com, {dirceutiegs, fernandosilgon, leonardosilvadm}@gmail.com

***Resumo.** Este artigo tem por objetivo expor uma metodologia de desenvolvimento de software, a Programação eXtrema, fazendo também uma comparação com o método de engenharia de software tradicional. Serão conceituados ambos os modelos citando seus pontos positivos e negativos.*

1. Introdução

Engenharia de software é a área voltada para a especificação, desenvolvimento e manutenção de sistemas de software aplicando práticas de ciência da computação e outras disciplinas, com objetivo de organização, produtividade e qualidade.

Já a Programação eXtrema é uma metodologia mais ágil para equipes pequenas e médias e que irão desenvolver softwares em constante mudança. Para isso, adota a estratégia de acompanhamento constante e realização de ajustes durante o desenvolvimento do sistema. “É uma disciplina de desenvolvimento de software.” (Beck, 2004, p. xiii)

2. eXtreme Programming

Todo o funcionamento da XP se baseia nessas três regras:

Valores: Simplicidade, Comunicação, Feedback, Coragem, Respeito.

Práticas: Jogo de Planejamento, Release, Metáfora, Design Simples, Testes de Unidade e Aceitação, Refatoração, Programação em Pares, Propriedade Coletiva do Código, Integração Contínua, Padrões de Codificação, Ritmo Sustentável e Cliente no Local.

Princípios: É o elo de ligação entre valores e práticas. Segue-se o princípio de valores para colocá-lo em prática. Sendo esse o fundamento da XP, só com a utilização incondicional dessas regras é possível utilizar essa metodologia.

A modelagem do banco segue os mesmos princípios que a programação, sua construção é feita mediante os dados que se tem de início, novas implementações são realizadas mediante a necessidade do sistema.

3. Engenharia de Software Tradicional

Concepção: Esta fase visa identificar o tipo de serviço de processamento de dados a ser executado (manutenção de um software existente ou a criação de outro), os objetivos a serem alcançados, recursos e prazos necessários para a execução do projeto.

Projeto Lógico: Especificar detalhadamente os elementos do software a nível lógico e tratar da especificação detalhada dos procedimentos externos ao computador.

Projeto Físico: Consiste em definir, dentro das restrições que possa existir, os seguintes pontos: organização do processamento (tempo real, batch); equipamento utilizado; o sistema operacional utilizado; suportes necessários; especificação dos programas do sistema; organização dos bancos de dados; controle do sistema; elaboração de *layouts*.

Programação: Desenvolver e simular o software especificado no projeto físico.

Implantação: Tem como objetivo o treinamento do usuário, a conversão/inicialização de arquivos e a implantação do software para produção.

Operação: Composta por duas subfases: Produção e Manutenção. Produção é responsável por manter o sistema funcionando e a manutenção é responsável por mudanças para mantê-lo viável.

4. Comparação entre Programação Extrema e Engenharia de Software

Os modelos atuais de desenvolvimento mais utilizados são baseados na Engenharia de Software. Há relativamente pouco tempo foi criado um novo método de desenvolvimento de software, que hoje é conhecido como XP (eXtreme Programming).

A seguir (Figura 1 e Figura 2) são apresentados os custos da mudança em cada um dos métodos de programação:

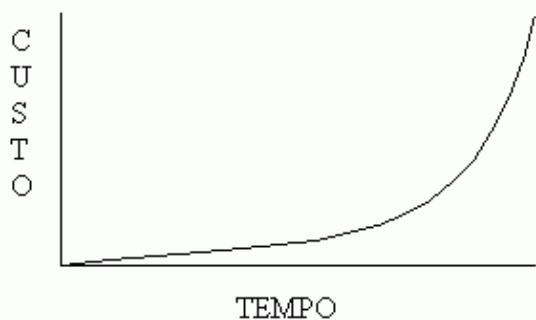


Figura 1: Utilizando Engª de Software

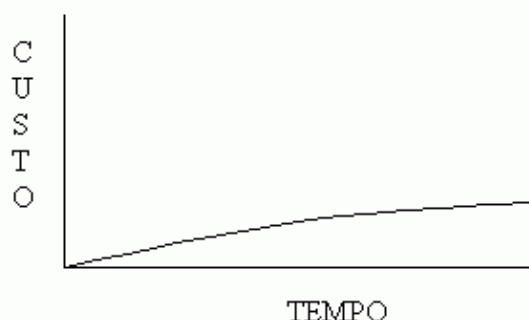


Figura 2: Utilizando Programação Extrema

De acordo com o gráfico, o software que foi criado utilizando a ACES (Abordagem Clássica de Engenharia de Software), quanto maior o tempo de vida, maiores serão os seus custos no momento em que passar por alguma mudança, enquanto que na XP os custos não são tão altos. Segundo Rocha (2006, p. 2), “mudanças nos requerimentos depois que a fase de análise e design foi concluída” seria o grande temor da ACES.

Tabela 1: Empresas que utilizam sistemas baseados no XP.

Empresa	Website
37signals	http://37signals.com
Odeo	http://odeo.com
IBM	http://ibm.com
Interlegis	http://www.interlegis.gov.br
Royal Bank of Scotland	http://www.rbs.co.uk
Google	http://www.google.com
Objective Solutions	http://www.objectivesms.com.br/
* Lembrando que nem todos os projetos dessas empresas utilizam XP	

5. Conclusão

Quando é implementado um sistema computacional utilizando ACES, é seguido todo o conceito deste. No término, temos um grande projeto e um grande volume de informações em manuais, dicionário de dados do sistema, entre outros, pois foi feito todo o levantamento de dados para que o sistema fosse completo e atendesse todas as necessidades do cliente. Desta maneira alguma alteração pode ser muito difícil de ser executada, pois muitas vezes é necessário reescrever boa parte do programa para se obter os resultados esperados. Pelo fato de o sistema ser grande, na maioria das vezes o cliente não utiliza tudo o que o sistema oferece.

Um dos pontos fortes do XP é o fato de que não é realizado todo um levantamento de dados para então iniciar o processo de desenvolvimento. O cliente está desde o início sempre presente e interagindo no projeto. Os testes são a primeira coisa a ser escrita e o feedback tende a ser rápido. Por este motivo é um modelo onde qualquer alteração é realizada sem muitas dificuldades. Neste projeto as pessoas se relacionam mais entre si, pois trabalham juntas em projetos diferentes ou em módulos diferentes do mesmo sistema. Apenas o que é solicitado pelo cliente é colocado em prática no programa evitando escrever desnecessariamente aquilo que não será utilizado.

Referências

- Beck, Kent. (2004). "Programação extrema explicada: acolha as mudanças". Porto Alegre: Bookman, 2004.
- Cunningham, Ward. (2001). "Manifesto for Agile Software Development", <http://agilemanifesto.org/>, Junho, 2006.
- Jeffries, Ronald E. (2006) "XProgramming.com - an Agile Software Development Resource", <http://www.xprogramming.com/>, Junho, 2006.
- Pressman, Roger S. (1995). "Engenharia de Software". São Paulo: Makron Books, 1995.

Rocha, Helder da. (2006) “eXtreme Programming”, <http://www.dc.ufscar.br/~rosangel/mds/Aula-09-XP/xp.pdf>, Agosto, 2006.

Tiegs, Dirceu P. (2006) “eXtreme Programming”, <http://dirceu.info/blog/?p=48>, Junho, 2006.

Wells, Don. (2006) “Extreme Programming: A Gentle Introduction.”, <http://www.extremeprogramming.org/>, Junho, 2006.

“Engenharia de software”, http://pt.wikipedia.org/wiki/Engenharia_de_software, Julho, 2006.

“Programação extrema”, http://pt.wikipedia.org/wiki/Programa%C3%A7%C3%A3o_extrema, Julho, 2006.