

**FACULDADE DE TECNOLOGIA DE AMERICANA**  
**FATEC - AM**

**PROGRAMAÇÃO ORIENTADA AO OBJETO EM EQUIPES**

**Elaborado por**

**WELLINGTON FÁBIO DE OLIVEIRA MARTINS**

**RA 071210**

**Orientado por**

**FRANCESCO ARTUR PERROTTI**

**AMERICANA**

**2009**

## Sumário

Resumo.....	3
Introdução.....	4
1 Trabalhando em Equipes.....	4
2 Programação Orientada ao Objeto.....	5
2.1 Definição de Objeto.....	5
2.2 Características que tornam a POO mais fácil de ser trabalhada em equipe.....	6
3 Princípios Básicos de Extreme Programming.....	7
3.1 Práticas do Extreme programming.....	7
4 Conclusão.....	9
Bibliografia.....	10

## Resumo

Com a finalidade de auxiliar o profissional da área de Tecnologia da Informação que acaba de se graduar e sai à procura uma vaga no mercado de trabalho, este artigo aborda algumas práticas de programação de sistemas de informação em equipe.

A popularidade da linguagem Java, da programação Web para servidores, de ferramentas de desenvolvimento como o IBM - RAD e Dot Net da Microsoft entre outras ferramentas, faz com que a Programação Orientada ao Objeto seja amplamente utilizada.

Serão abordadas neste artigo algumas práticas para desenvolvimento de softwares utilizadas e sugestões para trabalho em equipe, noções de Orientação a Objetos e alguns princípios de Extreme Programming.

**Palavras-chave:** UML – Unified Modeling Language

Extreme Programming – Metodologia de trabalho.

## **Introdução**

Uma boa forma de desenvolver sistemas trabalhando com equipes de programadores é utilizando a orientação a objetos.

As profissões que envolvem a Tecnologia da Informação, Análise de Sistemas, Processamento de dados, Ciência ou Engenharia da Computação são muito mais abrangentes do que seus títulos, dentro de empresas ou departamentos de tecnologia as funções dos programadores, analistas, administradores e operadores de banco de dados, sistemas operacionais (Mainframes), redes, domínios e segurança da informação acabam se confundindo e exigindo que o profissional de qualquer uma destas áreas tenha conhecimento e interação com todas as outras.

Serão apresentadas a seguir algumas noções de Programação orientada ao objeto, equipes XP e sugestões de práticas desta metodologia no desenvolvimento de sistemas em equipes.

## **1 Trabalhando em Equipes**

Observando os profissionais atuais das áreas de TI que é uma profissão nova, mas já estruturada e dividida hierarquicamente dentro dos departamentos das organizações empresariais, apesar de suas nomenclaturas serem modificadas constantemente, suas funções e atribuições como a própria “Tecnologia da Informação” que até a alguns anos ainda não tinha este título, sendo chamado apenas de CPD ou Departamento de Processamento de Dados, hoje já está bem contextualizada não só como título, mas como área fundamental e em algumas organizações e não só um departamento, mas até mesmo o único serviço de determinadas empresas.

Por esse motivo, o conhecimento de metodologias de trabalho e os recursos tecnológicos utilizados é de extrema importância.

Em um ambiente corporativo o programador deve estar preparado para decifrar códigos e corrigi-los ou acrescentar funcionalidades, raramente um programador começará o sistema desde o início e mesmo que isso aconteça à maioria das funcionalidades já estarão pré-programadas nos ambientes de desenvolvimentos atuais.

É neste contexto que o conhecimento de programação orientada a objetos é de grande ajuda.

## **2 Programação Orientada ao Objeto**

A Programação Orientada ao Objeto (Object-Oriented Programming) ou (OOP) foi criada no início da década de 70 e se origina da linguagem Simula (Simula Language), concebida na Noruega no início da década de 60 que foi criada para fazer simulações, como o próprio nome sugere. Esse tipo de programação possui como principal característica a semelhança com o mundo real a maior facilidade de abstração das rotinas diárias.

Hoje quando se ouve falar de Objetos, Classes, Métodos, o que vem a mente de um programador é linguagem Java que foi completamente desenvolvida para utilizar esse tipo de programação.

O que a POO trás realmente de vantagem em relação à programação estruturada é principalmente a reutilização de códigos, a facilidade de abstração, e a escrita em módulos.

### **2.1 Definição de Objeto**

Uma definição para objeto seria a de “um ‘ente’ ativo dotado de certas características que o tornam ‘inteligente’, a ponto de tomar certas decisões quando devidamente solicitado.” (Sanra Rahal Júnior, 2009).

## 2.2 Características que tornam a POO mais fácil de ser trabalhada em equipe

Para que uma linguagem seja considerada Orientada ao Objeto ela precisa suportar quatro conceitos básicos: abstração, encapsulamento, herança e polimorfismo.

**Abstração** é a habilidade de modelar características do mundo real do problema que o programador esteja tentando resolver. Passar para o programa em forma de entidades ou Classes as características de coisas, pessoas ou situações e interagi-las.

**Encapsulamento** é a capacidade da linguagem de isolar certos dados ou rotinas de forma que estes só sejam utilizados onde foram criados, um dado ou rotina envolvido por código só visível na rotina onde foi criado, o que possibilita ao programador, maior segurança quanto aos acessos indevidos por parte de outras rotinas, semelhante a variáveis locais definidas em linguagem C, Pascal e outras linguagens quaisquer.

**Herança** é principal característica que permite um código OO ser re-adequado, pois se já foi implementado em uma entidade pai inúmeras características (Atributos) estas podem ser reaproveitadas por outras entidades do mesmo tipo.

**Poliformismo** é a capacidade que um objeto tem de possuir inúmeros comportamentos, um mesmo código que pode ser aplicado a várias classes de objetos.

Estas são as principais características da OOP que fazem com que ela seja largamente utilizada na atualidade e juntamente com as práticas que serão apresentadas a seguir uma boa forma de programar em equipe.

### 3 Princípios Básicos de Extreme Programming

Segundo (Teles, 2005) uma definição para Extreme Programming seria: “Uma metodologia de desenvolvimento de software, nascida nos Estados Unidos ao final da década de 90 e que vem fazendo sucesso em diversos países, por ajudar a criar sistemas de melhor qualidade, que são produzidos em menos tempo e de forma mais econômica que o habitual”.

Os principais fundamentos do XP tiveram origem em meados da década de 80 por Kent Beck e Ward Cunningham então funcionários da Tektronix, Inc, mas só chegaram aos atuais conceitos quase uma década depois como um conjunto de boas praticas para programadores que na época que foram condensadas no padrão de linguagem *Episodes*.

A seguir serão apresentadas as principais práticas desta metodologia, cada um desses valores que são explicados de uma forma um mais abrangente no livro citado acima de Vinícius Manhães Teles, Mestre em Informática pela UFRJ.

#### 3.1 Praticas do Extreme programming

Uma das primeiras práticas observadas é a utilização de um ciclo de vida do sistema iterativo, mas com ciclos um pouco diferentes e mais curtos.

Neste artigo Serão apresentadas somente as práticas primárias.

##### 3.1.2 Ambiente Informativo

Para atingir os objetivos são utilizados diversos instrumentos tais como:

- Cartões com histórias colocados em um mural.
- Quadro branco.
- Gráficos com informações relevantes para a equipe.

- Post it sobre as paredes.
- Flip charts.

Cujo o principal objetivo como já percebido é o melhor desempenho da comunicação.

### 3.1.3 Ciclo semanal.

São realizadas reuniões semanais onde os desenvolvedores se reúnem com o cliente para priorizar um pequeno conjunto de funcionalidades que possam ser implementadas e testadas completamente naquela semana.

Nestas reuniões que recebem o nome de **Jogo do Planejamento**, o cliente tem o direito de informar as funcionalidades, também chamadas de **histórias** além de indicar a prioridade das mesmas, Enquanto os desenvolvedores podem estimar e apresentar suas considerações técnicas.

### 3.1.4 Histórias

São funcionalidades do sistema geralmente escritas em conjunto ou pelo próprio cliente no Jogo do planejamento é expresso em **cartões** o que o cliente deseja daquele módulo do sistema e também pode ser estimulado um **prazo** e uma **prioridade**.

Toda a iteração semanal é baseada nestes cartões que não precisam necessariamente ser concluídos em uma semana, mas sim de acordo com o prazo estipulado, eles são presos no mural e acompanhados diariamente pela equipe.

No mural os cartões podem ser separados entre não **iniciados**, **em andamento**, **finalizados**, e acompanhados facilmente por todos os participantes durante a semana.

### 3.1.5 Programação em par



Todos os códigos produzidos no projeto são desenvolvidos por duas pessoas juntas, diante do mesmo computador, revezando-se no teclado. Uma digita enquanto a outra revisa e corrige, essa dupla pode e deve sofrer rodízios freqüentemente, para não criar dependência.

“Quem trabalha continuamente com programação em par se habitua a corrigir e ter seu trabalho corrigido dezenas de vezes ao dia. A incidência de erros identificados pelo colega costuma ser tão elevada que surpreende quem não está acostumado ao uso da técnica” (Teles, 2005).

### **3.2 Trabalho organizado**

Para evitar inúmeros problemas como ter que fazer horas-extras para com freqüência para reduzir atrasos e tentar entregar o software em um prazo razoável.

Não se pode esquecer que programadores são seres humanos e se estressam aumentando assim as probabilidades de cometerem erros.

Deve haver muita organização das tarefas para evitar tais desgastes, muita atenção nas prioridades e prazos, por isso o ciclo semanal tem sido muito eficaz.

## **4 Conclusão**

Um projeto de Tecnologia da Informação geralmente só tem um “Início” e não um “Fim”, mas sim uma nova versão e outra em seguida e assim por diante. Uma equipe que inicia um projeto mesmo que este seja um sucesso, não será a mesma na próxima e na seguinte versão deste projeto. Muitos novos integrantes podem iniciar nesta equipe e continuar o trabalho.

Este artigo tem como objetivo demonstrar que uma boa forma de programação em equipes é a Programação Orientada ao Objeto aliada as práticas do Extreme Programming e um bom gerenciamento de requisitos, que permite que todas as etapas do projeto sejam executadas com maior probabilidade de êxito.

### Bibliografia

SANRA RAHAL JÚNIOR, Nelson Abu. **Artigo Científico** - Doutorando em Ciência da Computação (UFSC)

<[http://www.ccuec.unicamp.br/revista/infotec/artigos/leite\\_rahall.html](http://www.ccuec.unicamp.br/revista/infotec/artigos/leite_rahall.html)> Acessado no dia 27 de Abril de 2009.

TELES, Vinícius Manhães. **Extreme Programming: Aprenda como encantar seus usuários desenvolvendo software com agilidade e alta qualidade** – 2005.