

# Introdução à Programação eXtrema (XP)



Cursos de Verão 2008 - IME/USP

Mariana Bravo e Hugo Corbucci  
Departamento de Ciência da Computação

[www.agilcoop.org.br](http://www.agilcoop.org.br)

# O que é?

- XP é leve
- XP é focado no desenvolvimento de software
- XP funciona em times de qualquer tamanho
- XP se adapta à requisitos vagos e que mudam rapidamente

# De onde vem?

- Metodologia de desenvolvimento de software aperfeiçoada nos últimos 8 anos.
- Ganhou notoriedade a partir da OOPSLA'2000.
- Nome principal: Kent Beck.

# Os Valores de XP

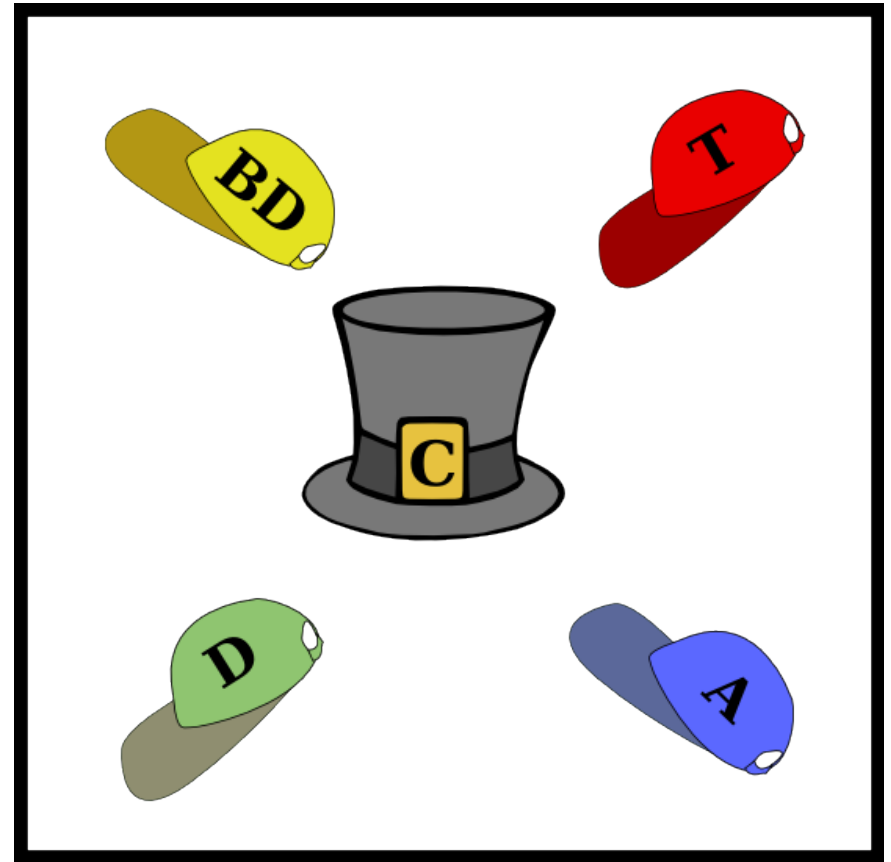
- Comunicação
- *Feedback*
- Coragem
- Simplicidade
- Respeito

# Introdução às práticas de XP

- Atividades a serem adotadas no cotidiano da equipe
- Adaptáveis a diferentes contextos
- Se suportam e complementam
- Permitem adoção em pequenos passos

# A equipe e seu ambiente

- Time completo:  
Equipe inteira com o mesmo objetivo
- Sentar juntos:  
Todos na mesma sala e próximos



# A equipe e seu ambiente

- Papéis

- *Coach*: Lembra a todos as práticas e ajuda com dificuldades na equipe
- *Tracker*: Mantem informações sobre o projeto e elabora gráficos que mostrem as mais importantes à equipe
- *Cliente*: Determina o que é mais importante, responde dúvidas dos programadores e toma decisões sobre funcionalidades.

# A equipe e seu ambiente

- Área de trabalho informativa:  
A sala de trabalho deve transmitir as informações essenciais pra equipe e ajudar a resolver problemas.
- Cliente presente:  
Dúvidas devem ser respondidas com o mínimo de atraso possível e com o máximo de contato.



# Jogo do planejamento

- Clientes escrevem e priorizam histórias
- Desenvolvedores estimam as mais prioritárias
- Suporte do *coach* e do *tracker* para evitar otimismo ou pessimismo excessivo

# No início

- Exploração:  
Antes do desenvolvimento do sistema, a equipe passa um período estudando e trabalhando com as tecnologias.
- Spikes:  
Ensaios de código rápidos com o intuito de aprender ou comunicar idéias.

# Tudo várias vezes

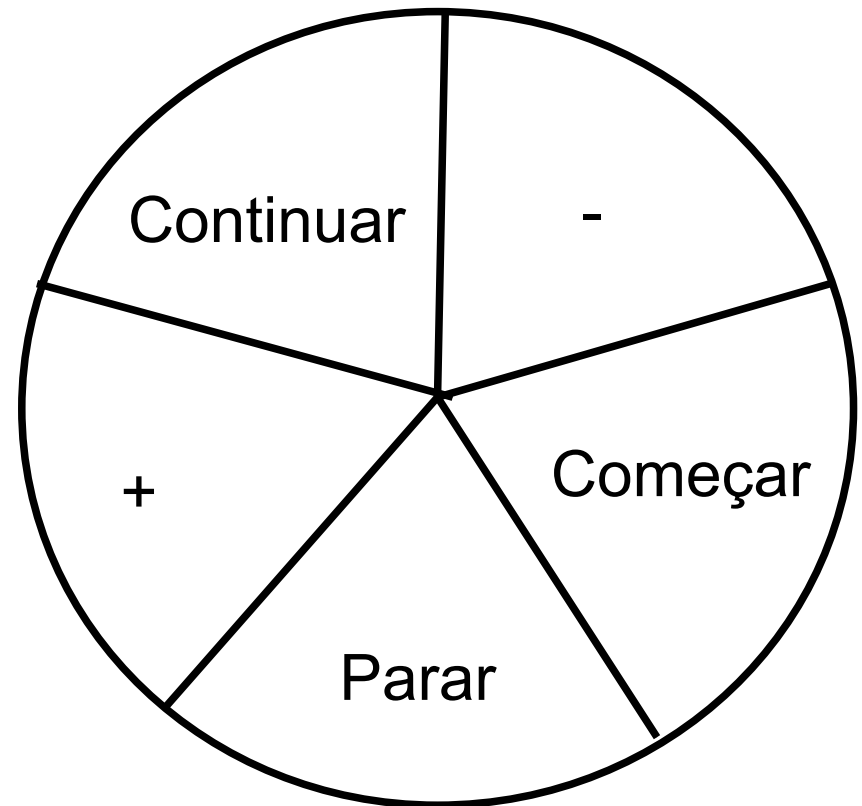
- Ciclos
  - diários:  
Alta granularidade e precisão
  - semanal:  
Granularidade média
  - da estação:  
Planejamento relativamente vago com folga

# Sem exagerar

- Trabalho energizado:  
Balancear a intensidade do trabalho para não desgastar a equipe.
- Folga:  
Não planeje até o último minuto. Deixe uma folga para imprevistos porque eles sempre surgem.

# Melhorando sempre

- Análise de causa inicial
- Retrospectiva:  
Nada é perfeito.  
Tudo sempre pode melhorar e, para isso, precisa entender o que deu certo e errado.

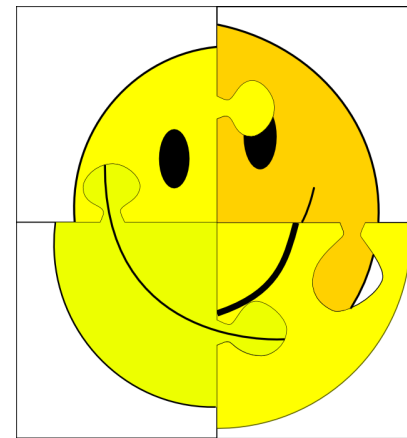
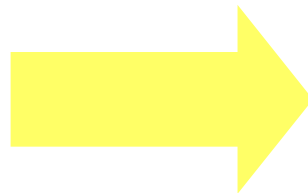
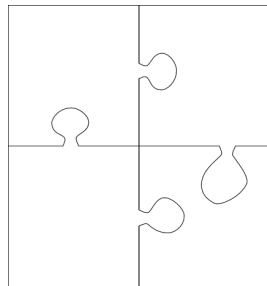


# Trabalhando em equipe

- Código compartilhado:  
Eu fiz, você arruma, nós nos ajudamos
- Padronização do estilo de código:  
Seu código e meu código devem ser quase idênticos e indistinguíveis

# Código unificado

- Repositório único
- Integração contínua



# Build de 10 minutos



Fonte:  
<http://xkcd.com/303/>



# Design Incremental

- Simples  
YAGNI: You Aint Gonna Need It  
“Você não vai precisar disso”
- Refatoração  
DRY: Dont Repeat Yourself  
“Não se repita”
- Investimento contínuo:  
Refatorar é um investimento pro futuro

# Testes

- Desenvolvimento dirigido por testes
- Testes automatizados
  - Unidade: Testa o código
  - Aceitação: Testa a funcionalidade
  - Interface: Testa a interação com usuário

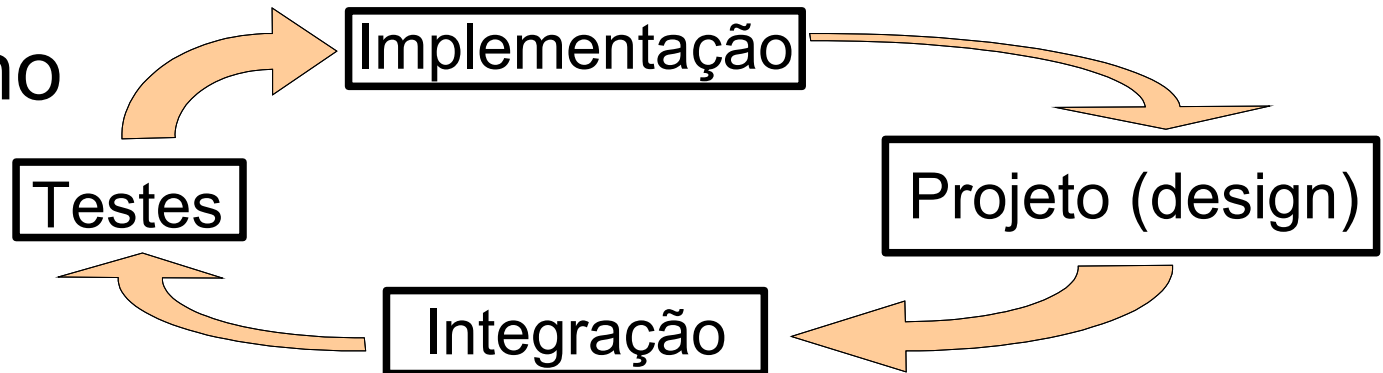
# Redundância

- Programação pareada:  
Distribuindo conhecimento na equipe



# Um dia de XP

- Escolhe uma história do cliente.
- Procura um par livre e um computador.
- Seleciona um “cartão de história”.
- Discute modificações recentes no sistema
- Discute história do cliente
- Entra no ciclo:



# Atos constantes no desenvolvimento

- Executa testes antigos.
- Busca oportunidades para simplificação.
- Modifica desenho e implementação incrementalmente baseado na funcionalidade exigida no momento.
- Escreve novos testes.
- Enquanto todos os testes não rodam a 100%, o trabalho não está terminado.
- Integra novo código ao repositório.

# Começando

- Pequenos passos:  
As práticas podem ser adotadas aos poucos.
- Escolha de *coach*:  
O cara experiente que ajuda a equipe.
- Adaptando:  
Refletir sobre objetivos e como atingí-los.

# Quando não usar

- Quando os valores de XP conflitam com os valores da sua empresa ou equipe.
- Muito cuidado com:
  - Sistemas críticos
  - Sistemas muito grandes
  - Enorme sistema legado

# Envolva-se

- Conferências:
  - Agile 2008 (<http://www.agile2008.org/>)
  - OOPSLA 2008 (<http://www.oopsla.org/>)
- Sites colaborativos:
  - Martin Fowler (<http://martinfowler.com/>)
  - Ron Jeffries (<http://www.xprogramming.com/>)
  - Brian Marick (<http://www.testing.com/>)
  - Ward Cunningham  
(<http://c2.com/cgi/wiki?ExtremeProgramming>)



# Maiores informações

- AgilCoop:
  - <http://www.agilcoop.org.br/portal>
- Outros:
  - <http://www.agilealliance.org/>
  - <http://www.xispe.com.br/>
  - <http://www.extremeprogramming.org/>
  - <http://www.agilealliance.com.br/>