

Aula 07

Pós-Graduação
em Gestão de
Sistemas de
Informação

Métodos ágeis de desenvolvimento de software

Métodos e Técnicas em
Engenharia de Software

Profa. Alexandra

Prof. Thiago

Objetivos

- Após esta aula, você deverá ser capaz de:
 - Compreender os princípios que regem as metodologias ágeis (Manifesto Ágil)
 - Descrever as práticas de planejamento e desenvolvimento adotadas pelo XP e SCRUM
 - Compreender as limitações e desafios à adoção de métodos ágeis

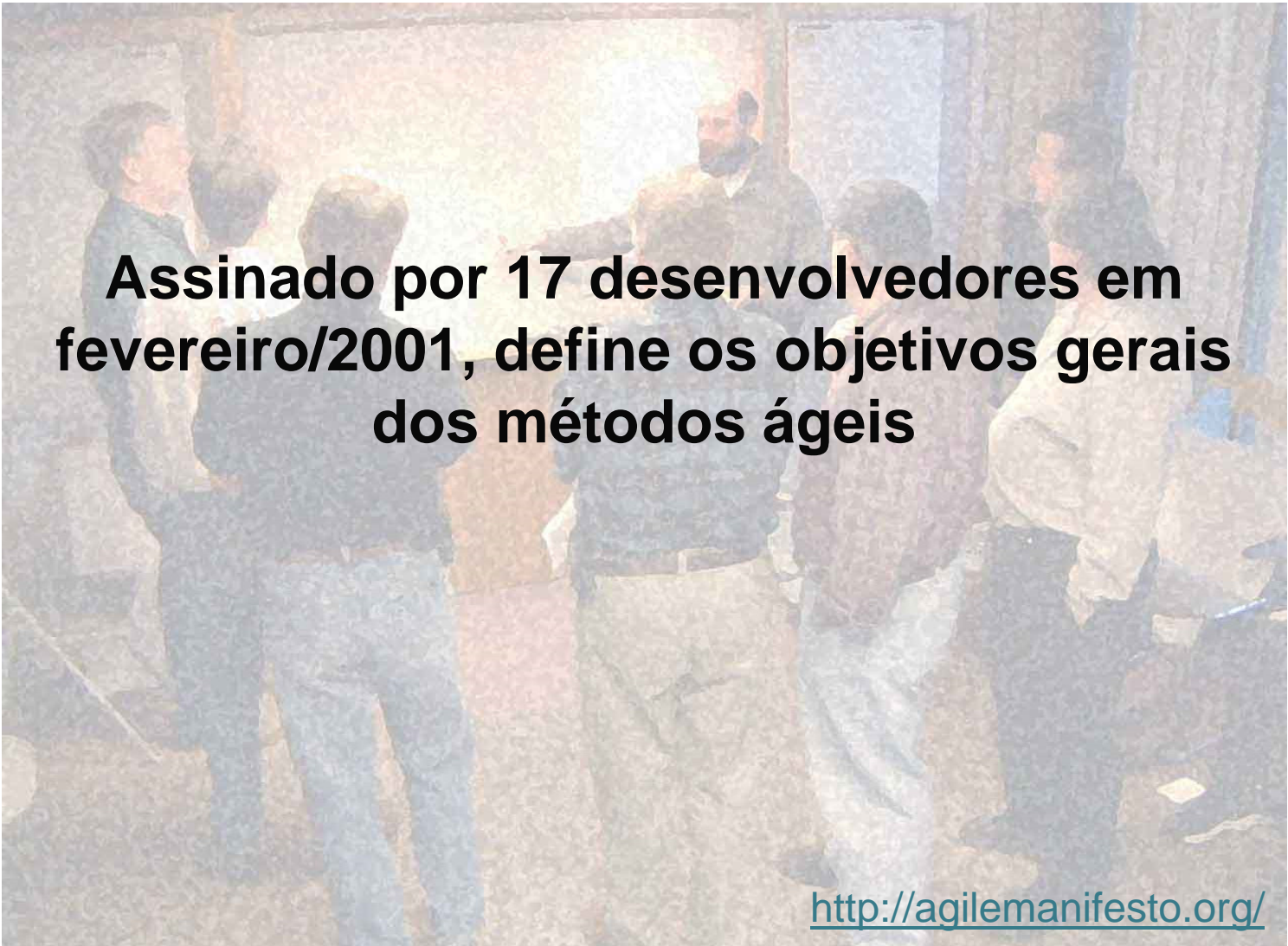
Métodos ágeis

Na última década vem surgindo uma nova categoria de metodologias para desenvolvimento de software, a dos **métodos ágeis**

Problema:

- Metodologias fortemente baseadas em uma especificação completa (e estática) do sistema não garantem necessariamente a qualidade do produto final
- Em situações onde os requisitos são vagos e mudam constantemente, essas metodologias podem dificultar a resposta à mudanças
- Em situações de pressão (p. ex.: grandes alterações em prazos reduzidos), a metodologia é abandonada, impactando na qualidade

O “Manifesto ágil”



Assinado por 17 desenvolvedores em fevereiro/2001, define os objetivos gerais dos métodos ágeis

<http://agilemanifesto.org/>



Indivíduos e interações mais que processos e ferramentas



**Software em funcionamento
mais que documentação abrangente**



**Colaboração com o cliente
mais que negociação de contratos**



**Responder a mudanças
mais do que seguir um plano**

Métodos ágeis

Mas, como responder rapidamente à mudanças?

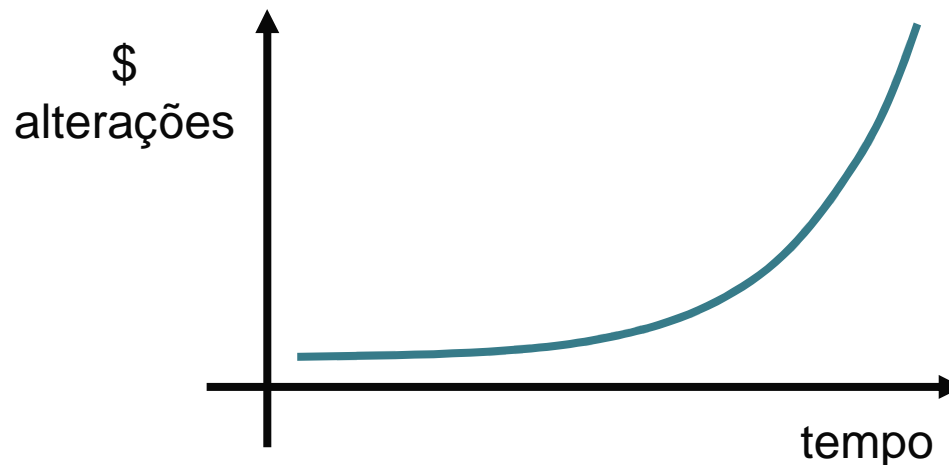
Atualmente, várias tecnologias aumentam a produtividade do desenvolvimento:

- **Orientação a objetos** → criação de componentes reutilizáveis de forma rápida
- **Padrões de projeto** → soluções padrão para problemas comuns
- **Refatoração** → qualidade do código (coesão e fraco acoplamento) melhora através de alterações metódicas, passo-a-passo
- **Testes automatizados** → verificação rápida do funcionamento do código após mudanças

Metodologias ágeis

Premissa das metodologias tradicionais:

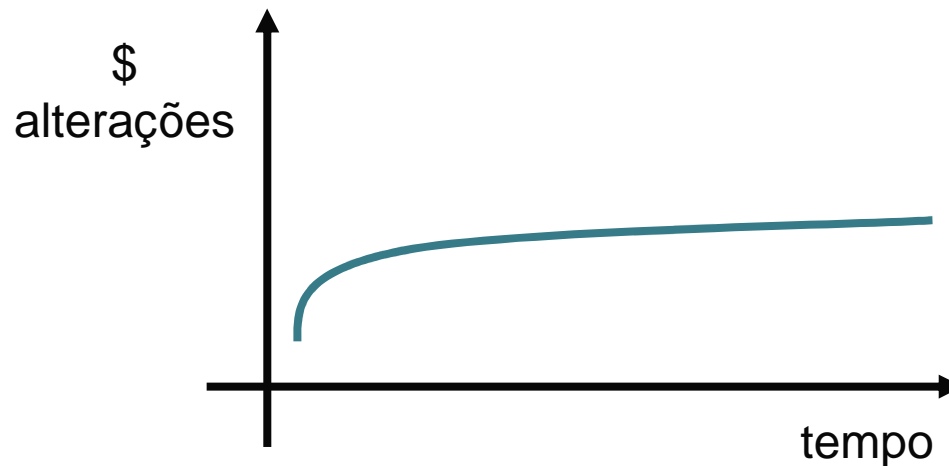
- Custo das alterações aumenta exponencialmente ao longo do tempo
- Logo, deve-se projetar tentando prever o que será necessário amanhã
 - Primeiro especificação completa de requisitos, depois modelagem completa, etc...



Metodologias ágeis

Premissa de XP

- A melhor produtividade trazida pelas tecnologias de desenvolvimento faz com que o custo de alterações não aumente ao longo do tempo
- Logo, devemos projetar *hoje* somente o que é necessário *hoje* (projeto mais simples possível)
- Alterações e mudanças serão possíveis sem uma penalidade muito alta em custo



Práticas de XP

- Programação em pares
- Projeto simples
- Refatoração
- Testes
- Entregas freqüentes
- Jogo do Planejamento
- Metáfora
- Propriedade coletiva
- Integração contínua
- Semana de 40 horas
- Cliente presente
- Padrões de codificação

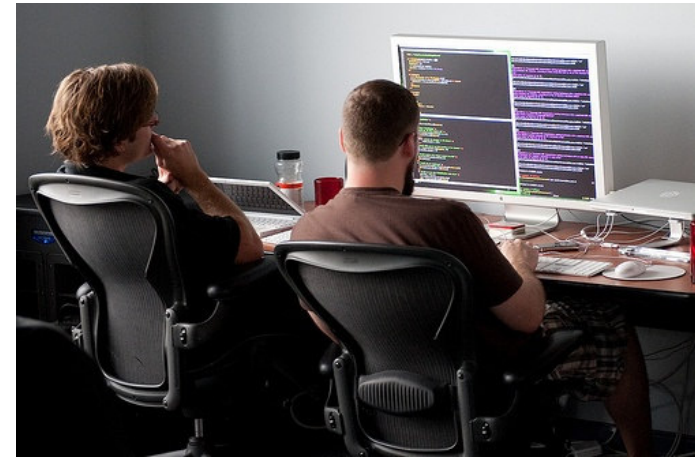


Principais práticas de XP

Programação pareada

Programação e definição de estratégia de projeto ao mesmo tempo

Aumento da produtividade



Projeto simples

A arquitetura do sistema deve implementar somente o que é necessário para as necessidades atuais do sistema

Principais práticas de XP

Refatoração

Permite a reutilização de código, generalização de funcionalidades, expansão das funcionalidades da arquitetura

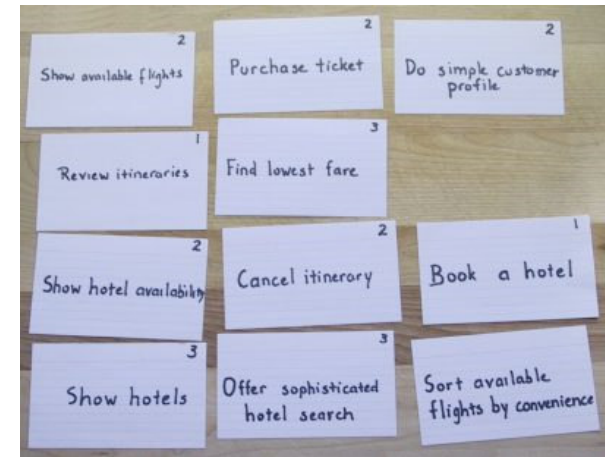
Testes

Com testes automatizados, aumenta a segurança em refatorar e modificar funcionalidades

Principais práticas de XP

Jogo do Planejamento

- O escopo de cada ciclo é baseado em prioridades de negócio e estimativas técnicas
- O planejamento é alterado com base nas dificuldades técnicas e nas estimativas de produtividade obtidas nos ciclos anteriores



Entregas freqüentes

- O ciclo de entregas é reduzido
- Cada entrega deve implementar alguma funcionalidade útil e verificável pelo cliente
- Permite correções de rumo do projeto o quanto antes

Principais práticas de XP

■ Metáfora

Em XP, as entidades técnicas do sistema devem ser definidas através de uma terminologia comum às equipe de negócios e desenvolvimento

- Ex: sistema de gerenciamento de contratos seria definido em termos de “contratos”, “clientes”, “endossos”

Objetivo semelhante ao da arquitetura de sistema, mas com foco na comunicação dos conceitos do sistema entre os envolvidos

Principais práticas de XP

■ Propriedade coletiva de código

Todos os desenvolvedores são autorizados a melhorar qualquer parte do código que julgarem necessário

■ Integração contínua

Todo o código é testado e integrado após algumas horas, com o apoio de testes automatizados

Principais práticas de XP

■ Semana de 40 horas

A produtividade da equipe é preservada definindo-se (e mantendo-se) planejamentos em função de uma semana de 40 horas

Regra de XP: horas extras por uma segunda semana seguida são sintomas de um problema no projeto que não pode ser resolvido com horas extras

Principais práticas de XP

■ Cliente presente

Um representante do cliente deve estar junto à equipe de desenvolvimento em tempo integral para responder questões e definir prioridades de menor escala

O representante deve ser um efetivo usuário do sistema a ser desenvolvido



Principais práticas de XP

■ Padrões de codificação

Como as duplas de desenvolvimento podem mudar constantemente e a propriedade do código é coletiva, deve existir um padrão de codificação

O padrão deve ser simples e adotado voluntariamente por toda a equipe

Planejamento ágil

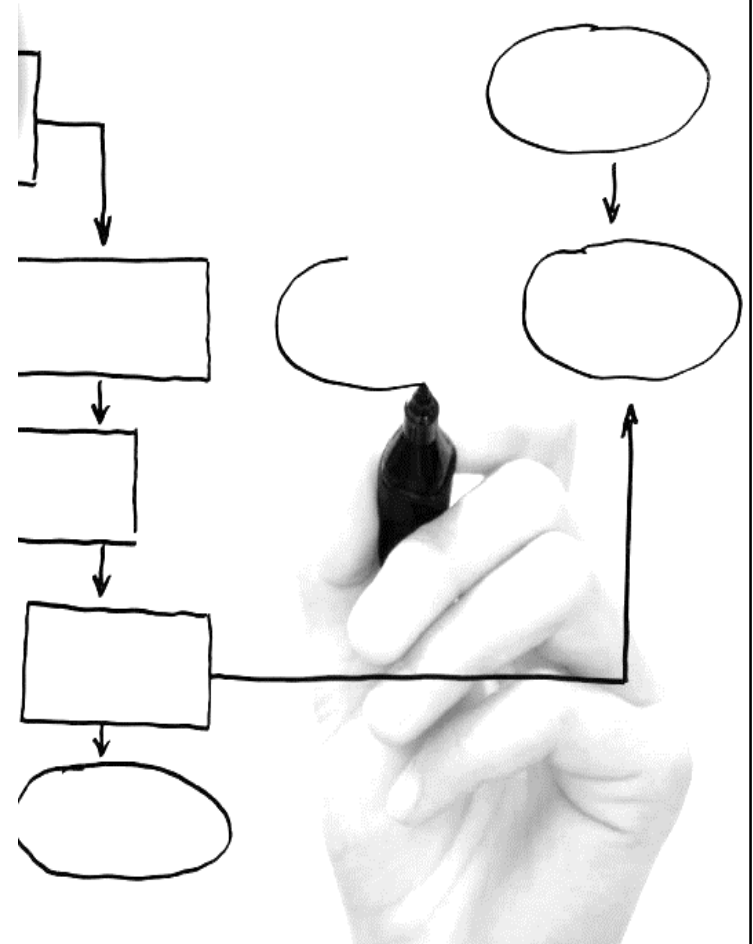
O planejamento nos métodos ágeis XP e Scrum funciona por meio de estimativas de **tamanho** baseadas em:

Pontos

- Mais associado ao Scrum, utiliza uma medida arbitrária em escala (ex: 1, 2, 3, 5, 8, 13...)

Dias ideais

- Mais associado ao XP, utiliza uma medida em dias (ou horas) “ideais” de trabalho (cerca de 50% do tempo total, excluindo o tempo para comunicação, ajuda, programação pareada)



Planejamento ágil

Pontos

- Discussões são dirigidas a funcionalidades
- É mais rápido
- A estimativa não perde a validade
- É uma medida pura de tamanho
- Meu dia ideal é diferente do seu
- Evita conversões precipitadas com horas de trabalho

Dias ideais

- Mais fácil de explicar para pessoas de fora da equipe
- É mais fácil para equipes com pouca experiência

Poker do Planejamento

- Técnica utilizada para estimar em pontos as histórias de usuário
- Formato de planejamento associado à metodologia Scrum
- Combinação de três técnicas:
 - **Opinião de especialista;**
 - **Analogia;**
 - **Desagregação.**

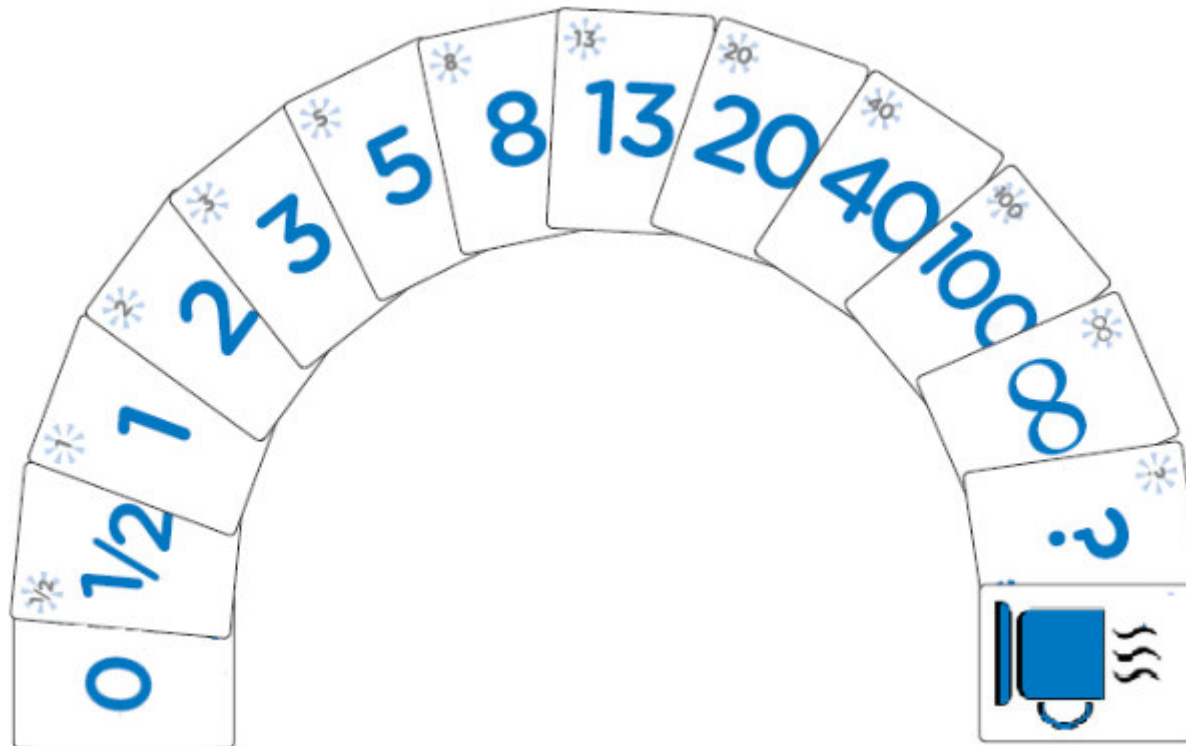
Participantes do Poker do Planejamento

- Equipe de Desenvolvimento (programadores, testadores, analistas, DBA);
- Cliente



Cartas do Poker do Planejamento

- Cartas com pontuação baseada na sequência de Fibonacci



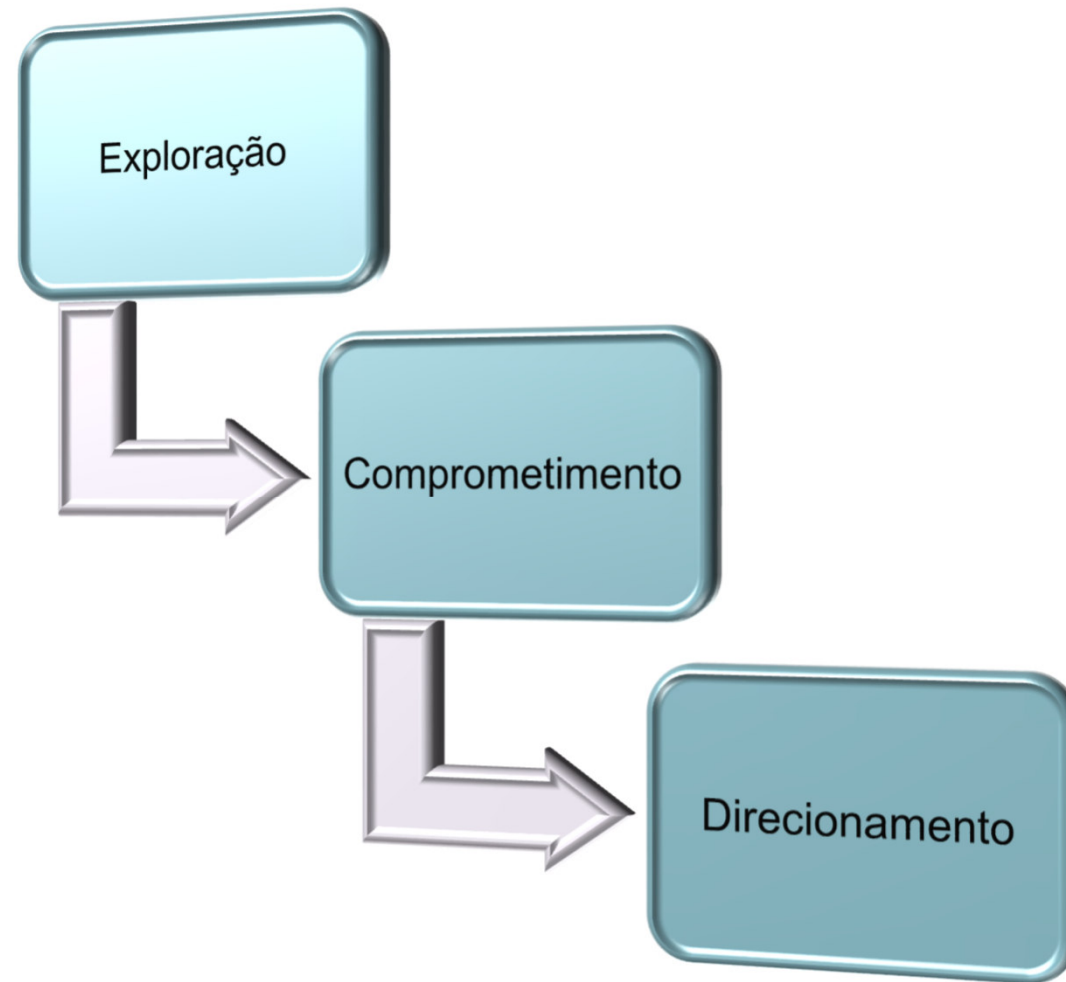
Jogo do Planejamento

- O Jogo do Planejamento é a atividade de definição dos objetivos do próximo ciclo do projeto
 - Lembre-se: uma das ênfases do projeto ágil está na habilidade de adaptação a mudanças
 - Logo, não faz sentido planejar detalhadamente para o longo prazo!
- Formato de planejamento associado à metodologia XP
- A definição do “jogo” abstrai o processo de planejamento para dois participantes
 - Negócios
 - Desenvolvimento

Jogo do Planejamento

- O objetivo do jogo é maximizar o valor do software produzido, minimizando o custo de desenvolvimento e o risco envolvidos
- A estratégia dos participantes é:
 - **investir o mínimo possível;**
 - **para colocar as funcionalidades mais valiosas em produção o mais rápido possível**
 - O investimento é feito conjuntamente com as técnicas de programação e de projeto criadas para reduzir o risco
 - As funcionalidades mais valiosas serão identificadas pela equipe de negócio com base no feedback obtido no ciclo anterior

Fases do Jogo do Planejamento



Fases do Jogo do Planejamento

- **Exploração:** descobrir novas coisas que o sistema pode fazer;
- **Comprometimento:** descobrir que subconjunto de todos os possíveis requisitos será explorado a seguir;
- **Direcionamento:** guiar o desenvolvimento em função dos fatos que ocorrem no seu andamento

Velocidade

- É a quantidade de trabalho que a equipe consegue fazer em uma iteração
- Geralmente é medida em pontos de história de usuário (Scrum) ou dias ideais (XP)
- Na primeira iteração a velocidade é um “chute”
- Usada para planejamento de iteração

Requisitos ágeis

- Em geral expressos por **histórias de usuário**
- Um cartão de história é a descrição, do ponto de vista de negócios, de uma funcionalidade ou alteração desejada

Como um (ator) eu quero / preciso de / devo / gostaria
de (executar uma ação) para (obter uma
funcionalidade)

Exemplo de História de Usuário

Como um vendedor responsável pelo setor de livros
eu quero procurar por livros filtrando por nome
para verificar se um dado livro está disponível
para pronta entrega.

Algumas tendências

- XP Distribuído
- Dívida técnica