

# Scrum no Desenvolvimento de Jogos

Metodologias e Tecnologias

Slides por:

Rafael Miranda Lopes  
William Quelho Ferreira

# Conteúdo

→ Metodologias:

- ◆ Desenvolvimento tradicional
- ◆ Desenvolvimento ágil

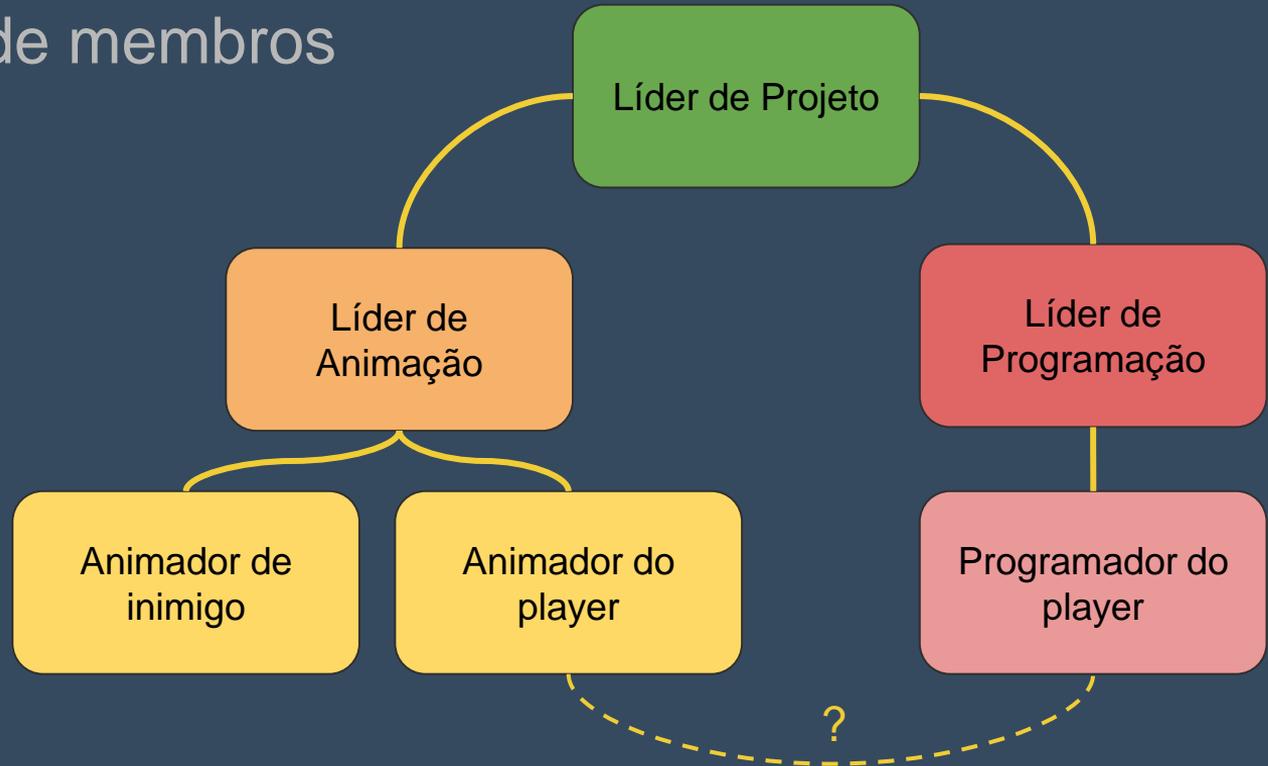
→ Scrum

# Processo de desenvolvimento?

1. Pensar numa ideia
  2. Planejar implementação
  3. Implementar
  4. Debugar
  5. Publicar
- Conhecido como “modelo cascata” ou desenvolvimento tradicional

# Problemas do Cascata

→ Hierarquia de membros



# Problemas do Cascata

- Etapas bem definidas e sem volta
  - ◆ Problemas com incerteza



# Problemas do Cascata

Focar em:

- Processos e ferramentas
- Documentação abrangente
- Negociação de contratos
- Seguir um plano

# Manifesto Ágil

Valorizar mais:

- Pessoas e interações
- Software em funcionamento
- Colaboração com o cliente
- Responder a mudanças

Fatores do outro slide são importantes, mas os deste são mais

# Desenvolvimento Ágil

Início

Protótipo

Potencial produto ideal

# Desenvolvimento Ágil

Início

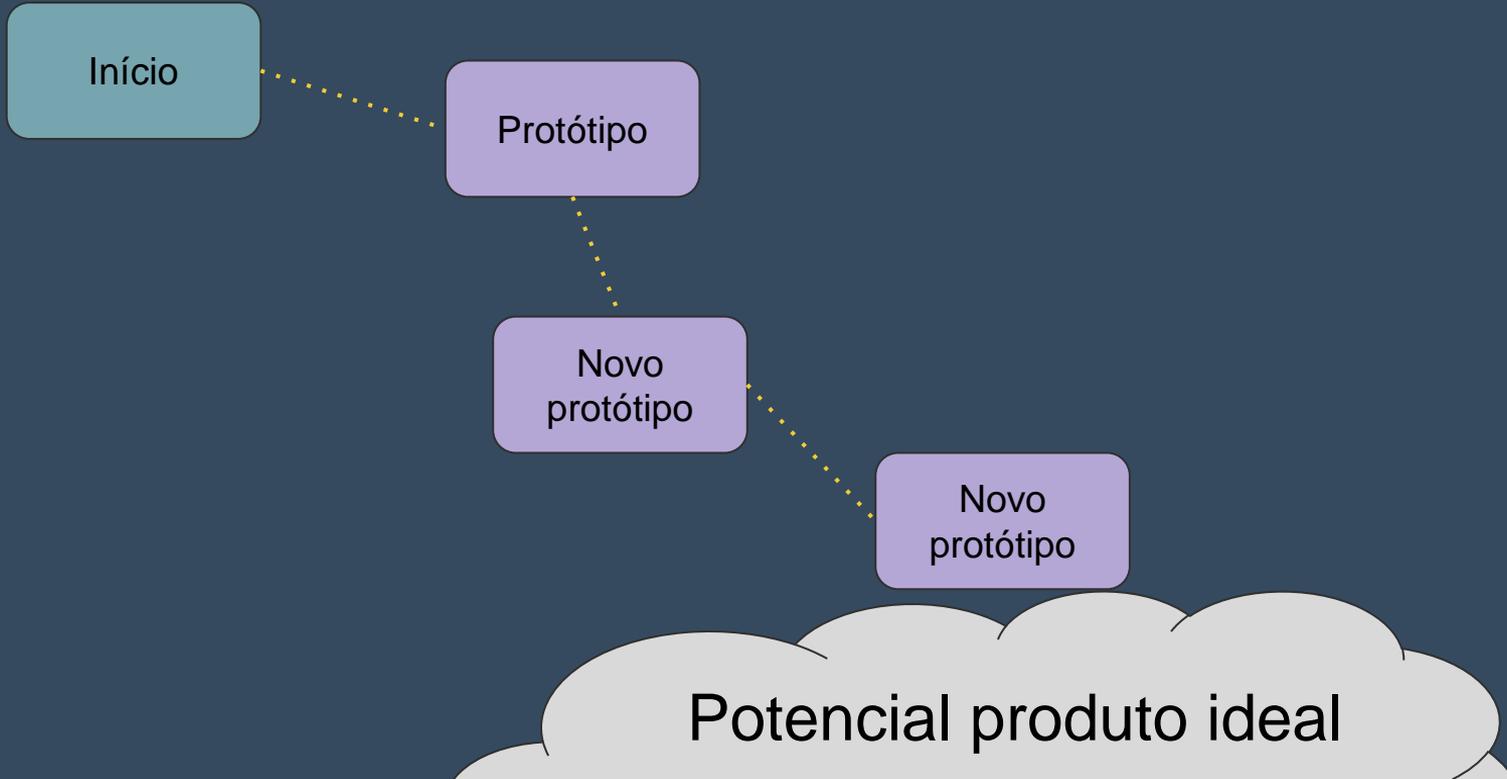
Protótipo

Novo  
protótipo

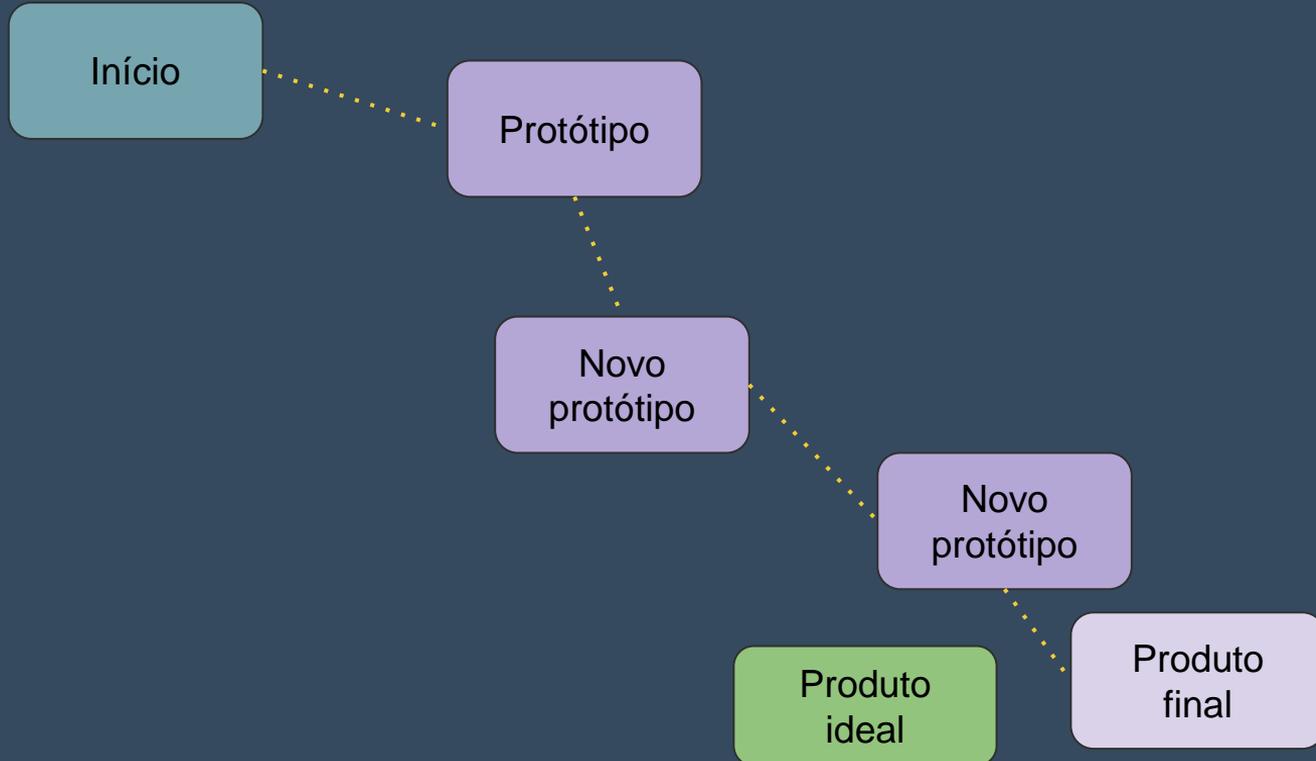
Potencial produto ideal

```
graph TD; Início --> Protótipo; Protótipo --> Novo_protótipo[Novo protótipo]; Novo_protótipo --- Potencial_produto_ideal((Potencial produto ideal));
```

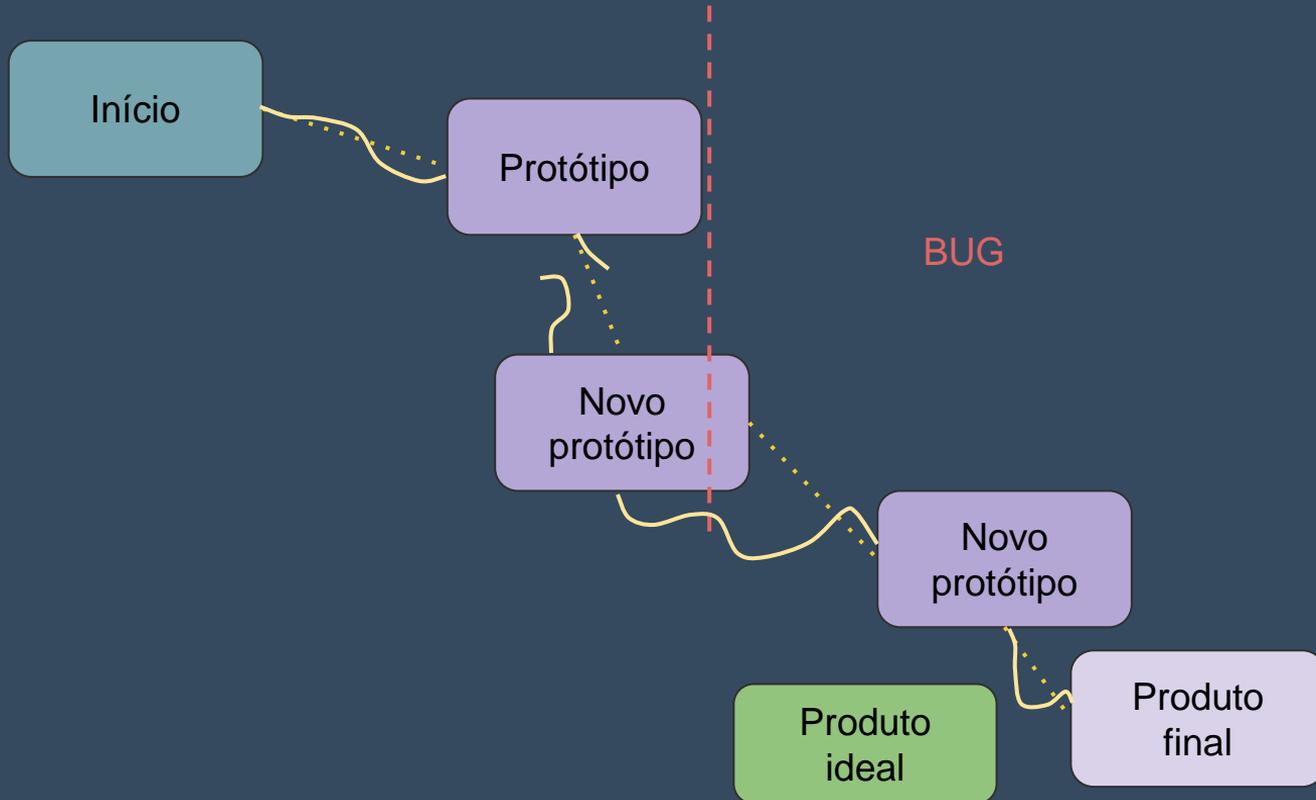
# Desenvolvimento Ágil



# Desenvolvimento Ágil



# Desenvolvimento Ágil

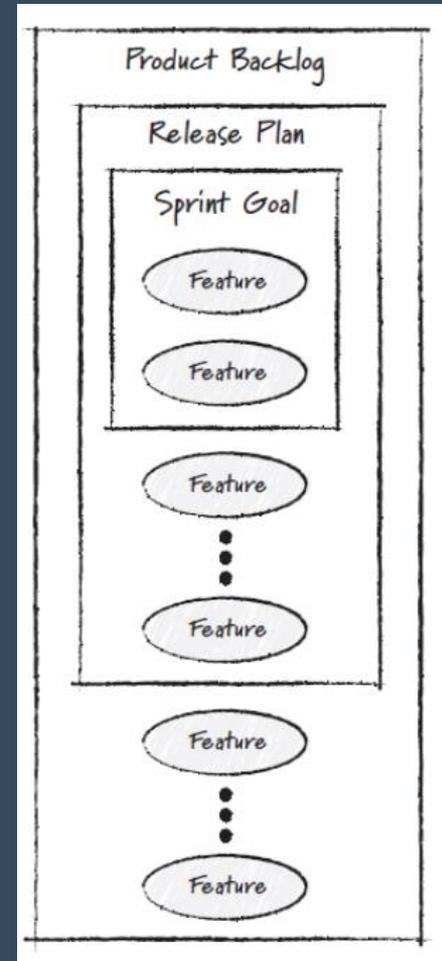
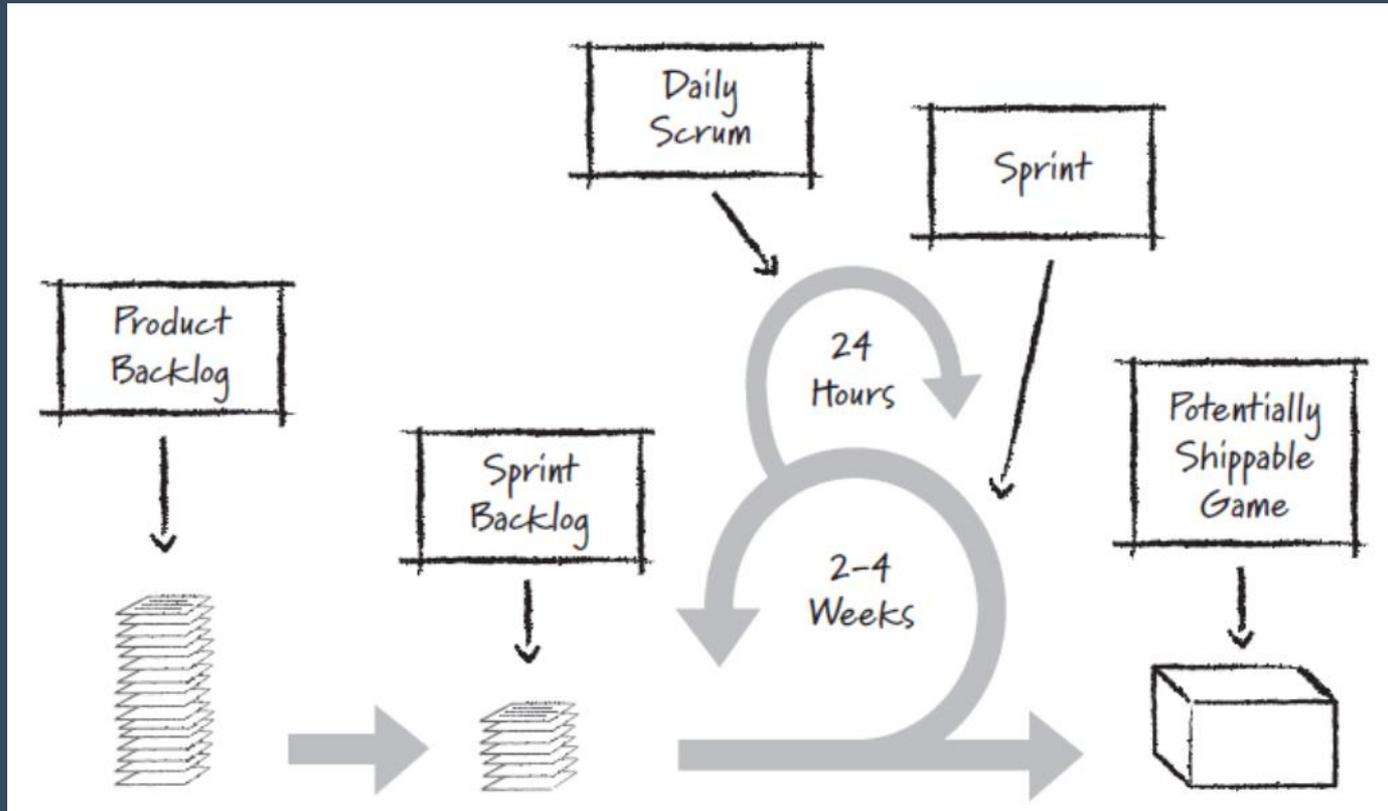


# Scrum

- Um método de desenvolvimento ágil
- Subdivide grupos em papéis diferentes
  - ◆ Sem hierarquia

# Agile Game Development with Scrum

- Clinton Keith



# User Stories

- Requisitos do ponto de vista de um usuário
- Deve tanto adicionar quanto mostrar valor
- Não entram em detalhes

# User Stories

- “Como <usuário>, quero <objetivo> [para que <motivo>]
- ◆ *Usuário*: quem se beneficiará da user story
  - ◆ *Objetivo*: uma feature, funcionalidade ou ferramenta
  - ◆ *Motivo*: o benefício em si trazido pela user story ao usuário quando a feature é utilizada (opcional)

Como jogador, quero poder pular para me movimentar verticalmente

# Sprint

- Cada iteração do software
- Tem duração predefinida (“*timeboxed*”)
- Dividido em:
  - a. Reunir grupo completo
  - b. Priorizar backlog de produto
  - c. Montar backlog de sprint
  - d. Repetidos daily scrums
  - e. Sprint review
  - f. Retrospectiva

# Product Backlog

→ Conjunto de todas as user stories do produto

O jogador  
pode pular

O jogador  
pode atirar

O jogador  
pode andar

Ambientação  
futurista

Chefes são  
desafiadores

Controles  
responsivos

Player aprende  
poderes dos chefes

Gameplay  
rápida

Variedade de  
inimigos

# Priorização de User Stories

→ Valor

→ Custo

→ Risco

→ Conhecimento

# Backlog de sprint

- Priorizar elementos do backlog de produto
- Subdividir cada elemento em tarefas específicas

O jogador  
pode pular

Animador: Criar animações de pulo  
Estimativa: 10h

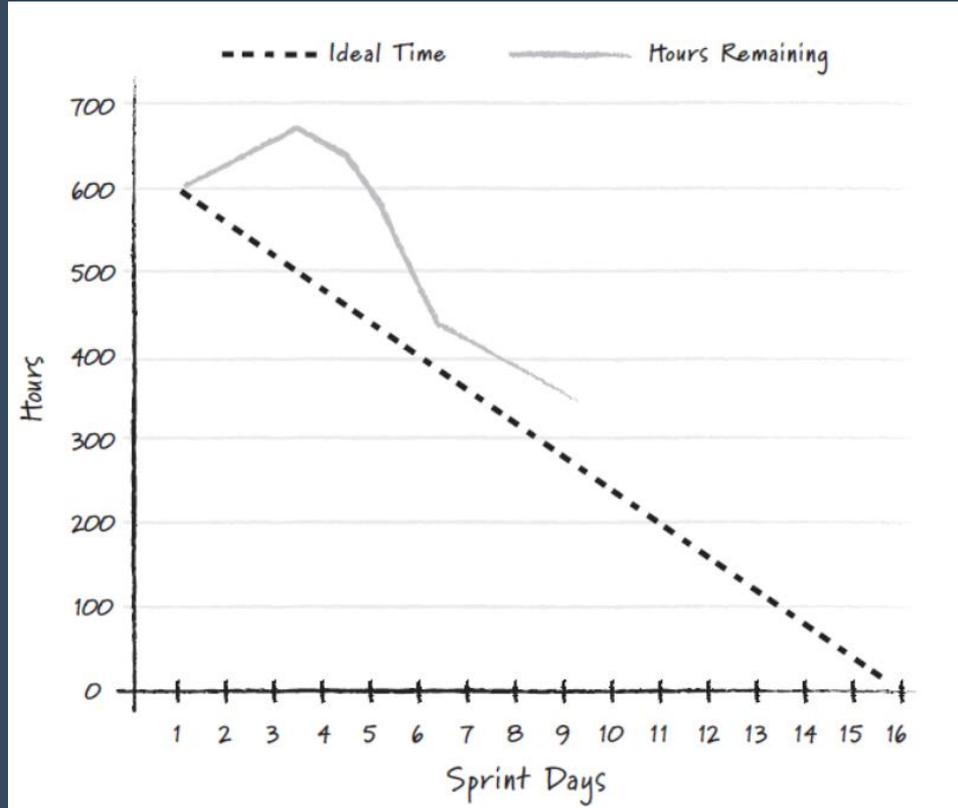
Programador: Implementar controle de pulo  
Estimativa: 5h

Programador: Implementar física de pulo  
Estimativa: 16h

Designer: Balancear o pulo  
Estimativa: 6h

Número de tarefas é baseado  
no tempo de sprint

# Burnout



# Daily Scrum

- Reunião rápida, de 5 a 10 minutos
- “O que eu fiz desde o daily scrum anterior?”
- “O que farei até o próximo daily scrum?”
- “Quais são os impedimentos que estão me atrasando?”

Não é uma reunião para solução de problemas

# Sprint Review

- Último dia de sprint
- Testar o jogo e avaliar o resultado atingido
- Casual
- Feedback honesto

# Retrospectiva

- Melhorar continuamente como o time adiciona valor ao jogo
- “O que estamos fazendo bem e devemos continuar a fazer?”
- “O que devemos parar de fazer?”
- “O que devemos começar a fazer?”

# Divisão do time

## → Desenvolvedores

- ◆ Reportar impedimentos
- ◆ Manter comprometimento
- ◆ Colaborar com colegas
- ◆ Buscar melhoria contínua

## → Scrum Master

- ◆ Assegurar que impedimentos sejam tratados
- ◆ Monitorar progresso
- ◆ Encorajar melhoria contínua

# Referências

→ Keith, C. *Agile Game Development with Scrum*. Addison-Wesley, 2010