Scrum

O desenvolvimento ágil no desenvolvimento de jogos

Slides por:
Giuliano Lourençon (giulianolourencon@usp.br)
Leonardo Tórtoro Pereira (leonardop@usp.br)
Rafael Miranda Lopes (rafael.miranda.lopes@usp.br)
William Quelho Ferreira (william.quelho.ferreira@usp.br)

Este material é uma criação do
Time de Ensino de Desenvolvimento de Jogos Eletrônicos (TEDJE)
Filiado ao grupo de cultura e extensão Fellowship of the Game (FoG), vinculado ao ICMC - USP

Este material possui licença CC By-NC-SA. Mais informações em:
https://creativecommons.org/licenses/by-nc-sa/4.0/

Processos de desenvolvimento de software

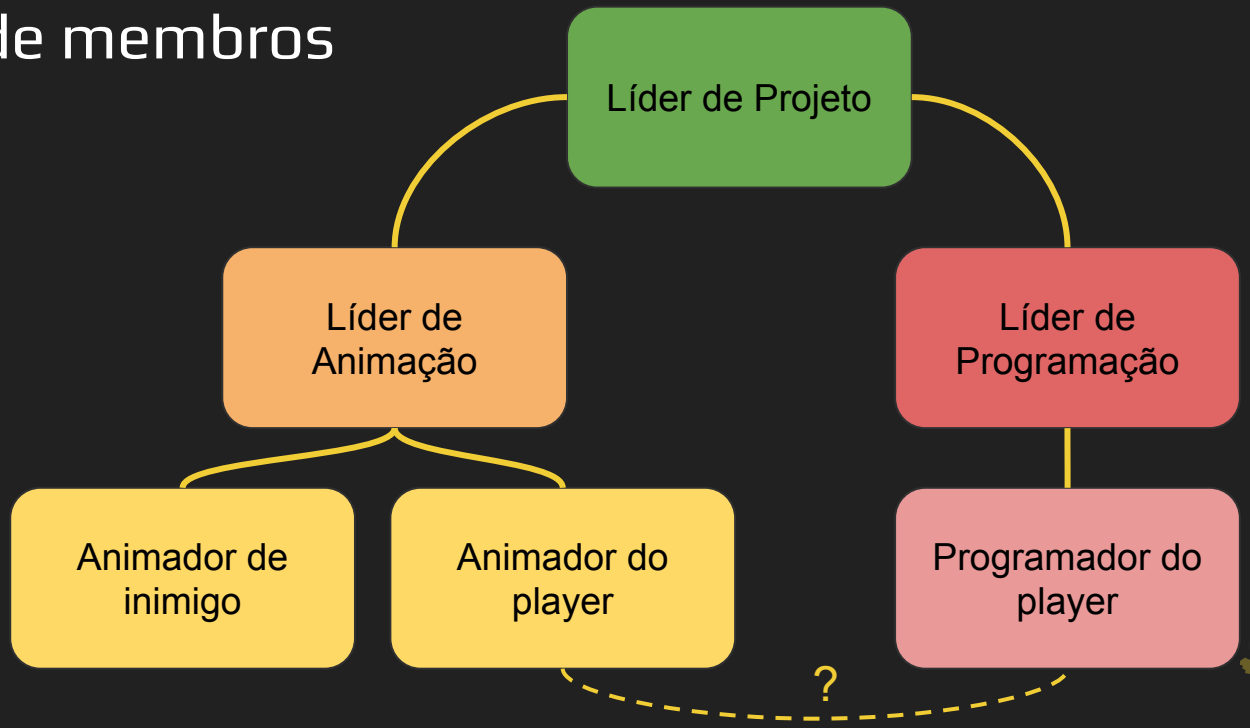
Processo Básico

1. Pensar numa ideia
2. Planejar implementação
3. Implementar
4. Debugar
5. Publicar

→ Conhecido como “modelo cascata” ou desenvolvimento tradicional

Problemas do Cascata

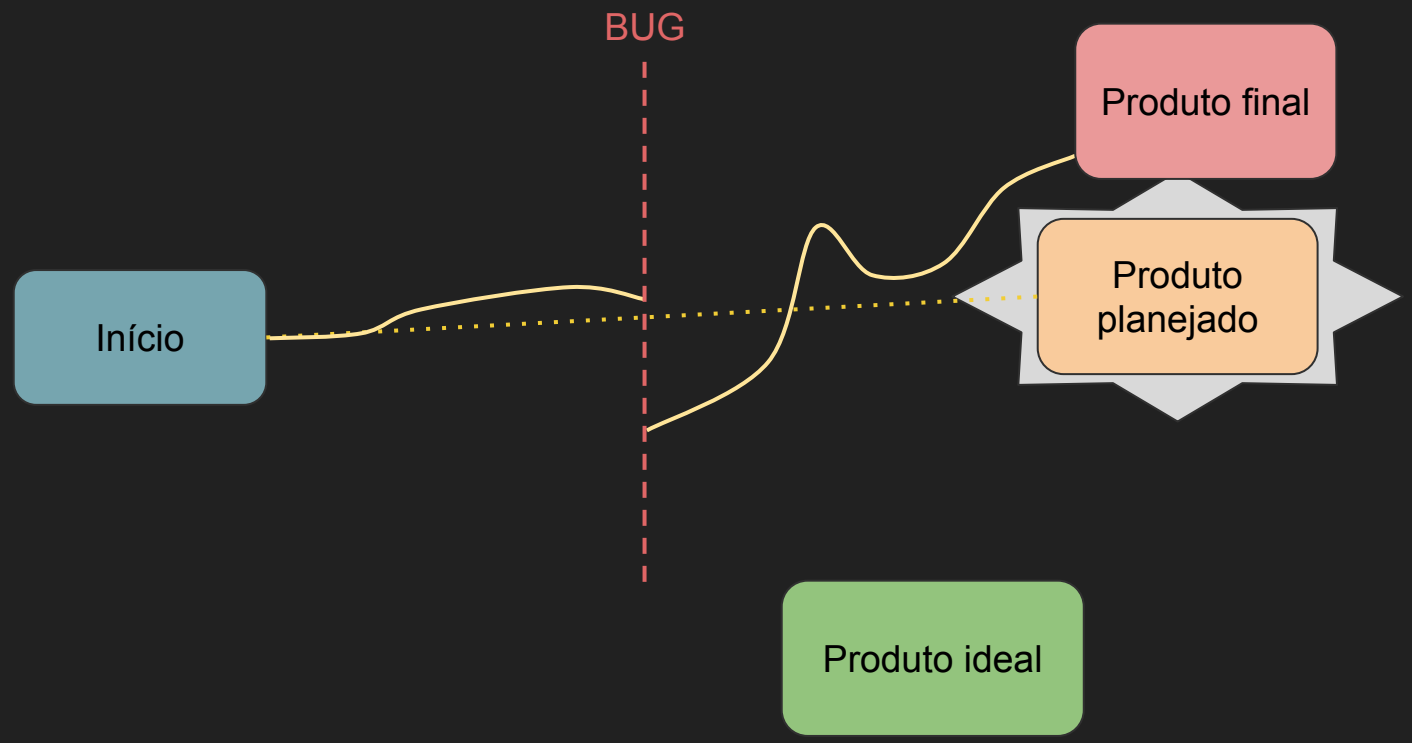
- Hierarquia de membros



Problemas do Cascata

- Etapas bem definidas e sem volta
  - Problemas com incerteza
  - Feito baseado em métodos de produção de engenharia
  - Uma boa coleta de requisitos pode ser difícil de ser feita

Problemas do Cascata



Problemas do Cascata

Focar em:

- ➔ Processos e ferramentas
- ➔ Documentação abrangente
- ➔ Seguir um plano

MODELOS ÁGEIS

Manifesto Ágil - 2001

- **Indivíduos e interações** são MAIS IMPORTANTES do que processos e ferramentas
- **Software funcionando** é MAIS IMPORTANTE do que a documentação completa e detalhada
- **Colaboração com o cliente** é MAIS IMPORTANTE do que a negociação de contratos
- **Adaptação às mudanças** é MAIS IMPORTANTE do que seguir um plano inicial

SCRUM

- **Framework** no qual podem ser aplicados vários métodos e técnicas
- Melhoramento contínuo:
  - Do produto
  - Da equipe
  - Do ambiente de trabalho

Pilares do SCRUM

- TRANSPARÊNCIA
  - A equipe deve conhecer o estado atual do projeto
- INSPEÇÃO
  - Dos artefatos e do progresso do projeto
  - Sem atrapalhar o desenvolvimento
- ADAPTAÇÃO
  - Do processo ou do produto

Pontos principais

- Times pequenos
- Cortar tudo que não é útil
- Começar o desenvolvimento o mais cedo possível
- Entregar versões funcionais o mais cedo possível
- Documentação em excesso é perda de tempo
  - E só gera dor de cabeça
- Documentação nunca vai ser perfeita antes de começar

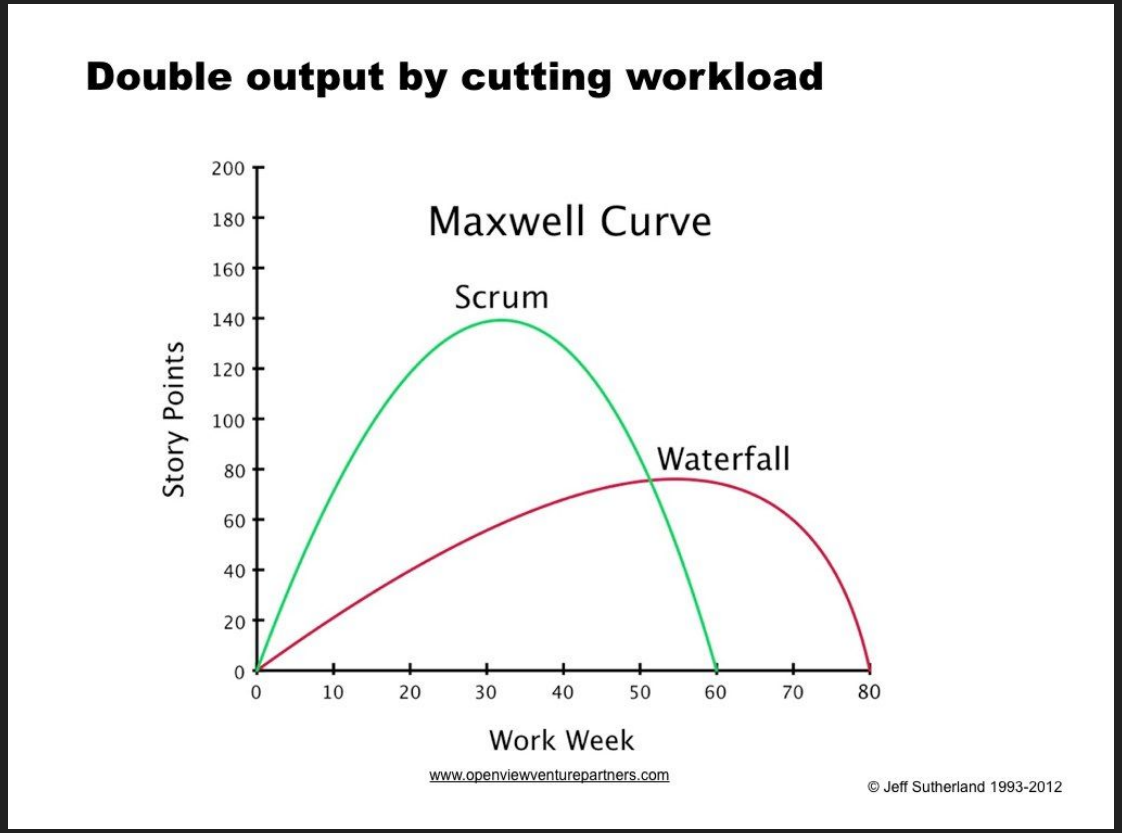
Pontos principais

- Desenvolver algo por meses ser ver resultado algum é entediante e desanimador
  - Abre espaço para que quando finalmente tiver algo pronto, nada seja como você e seu cliente imaginavam
- Cortar reuniões para o mínimo possível!
  - “Poderia ter sido um email”
  - Somente o essencial e SEMPRE objetivo
  - Mais tempo pra jogar :)

Por que o Scrum?

Por que o Scrum?

- ➔ Ótima pergunta! Devemos ser céticos sobre tudo :)
- ➔ Estudos de caso extremamente positivos
- ➔ Desenvolvedores entregam muito mais em muito menos tempo



Comparação da produtividade do Scrum com o método Cascata

Fonte:

https://34slpa7u66f159hfp1fhl9aur1-wpengine.netdna-ssl.com/wp-content/uploads/2013/02/Maxwell-Curve.jpg

Por que o Scrum?

- Um dos maiores ganhos de tempo é seguir a filosofia:
  - "A tarefa só acaba quando ela estiver pronta"
- Isso reduz tempo perdido re-aprendendo código antigo que não funcionava corretamente e foi deixado pra lá
- Nosso cérebro não consegue armazenar algo complexo como código por tanto tempo

Por que o Scrum?

- É até 20x mais rápido terminar tudo antes de mudar de tarefa!
- Arrume todos os bugs, detalhes de funcionalidade, etc.
  - “Faça tudo certo de primeira!”
- De novo, mais tempo para ver séries e animes :)

Por que o Scrum?

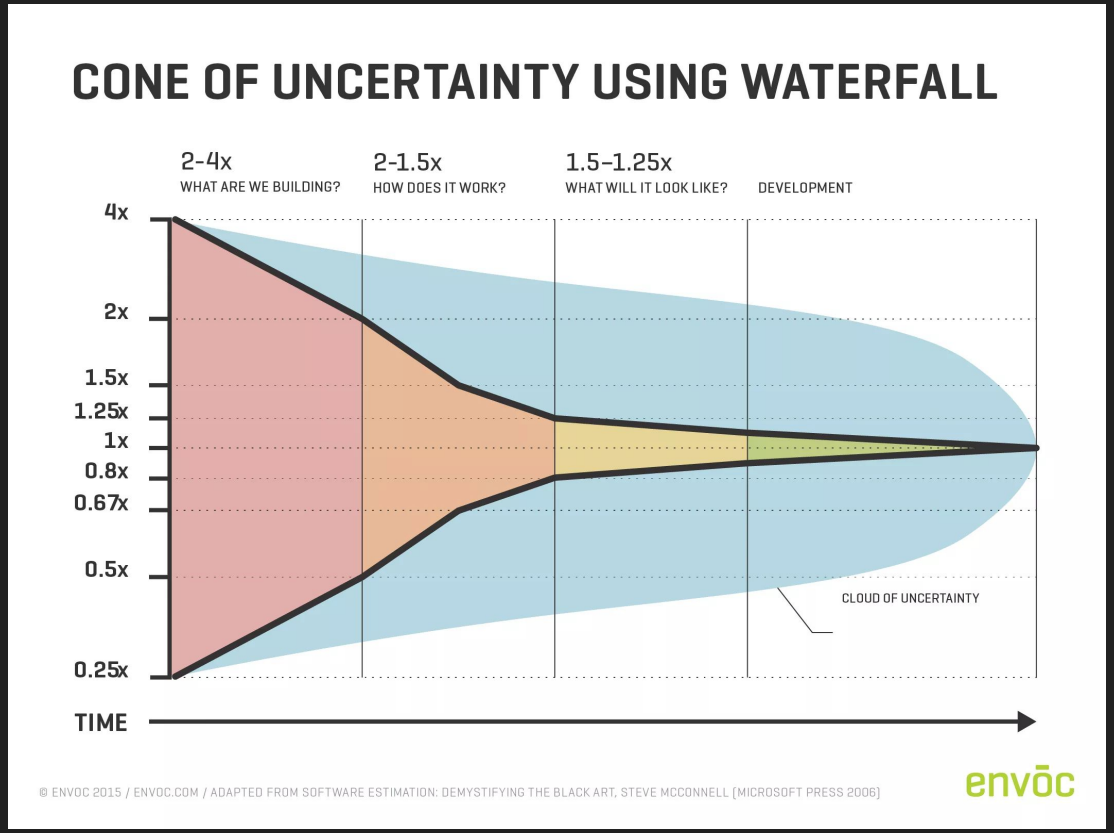
- Reduz incerteza do projeto e gasto de tempo com planejamento inútil*!
- *Planejamento é extremamente importante!
  - Exageros na documentação, não.

Por que o Scrum?

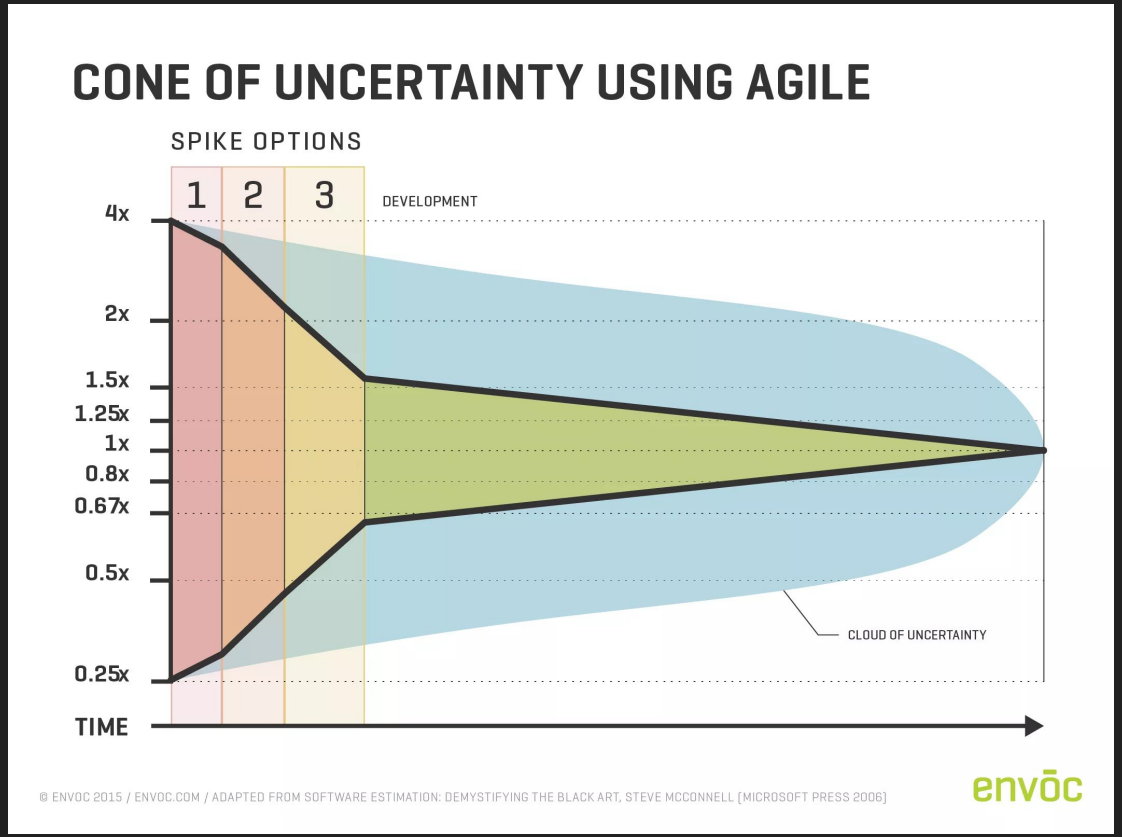
- "O mapa não é o terreno"
  - Não importa o quão bem você conheça o projeto nem o quão experiente você se considera:
    - É impossível planejar perfeitamente um projeto antes de começar!
- Na situação hipotética que você, grande deus do dev, consiga, sua equipe não. Eles vão errar.
- E não adianta fazer algo que só funciona para você :)

Por que o Scrum?

- ➔ Qualquer planejamento no começo de um projeto pode errar o tempo para 4x a mais ou a menos que o tempo real.
- ➔ A margem de erro decresce ao longo do desenvolvimento



Cone da incerteza usando o modelo Cascata.
Fonte: https://i1.wp.com/envoc.com/uploads/files/EstimateVariability-01.png



Cone da incerteza usando um modelo Ágil (do qual o Scrum faz parte)
Fonte: https://i1.wp.com/envoc.com/uploads/files/EstimateVariability-03.png

Por que o Scrum?

- A redução da incerteza do scrum vem principalmente da redução da documentação detalhada
- Sua maior documentação será o backlog do produto
  - Conjunto de histórias
  - Começam abstratas
  - Conforme surge certeza de como implementá-las, são atualizadas com versões mais concretas
  - Cada uma tem sua prioridade

Por que o Scrum?

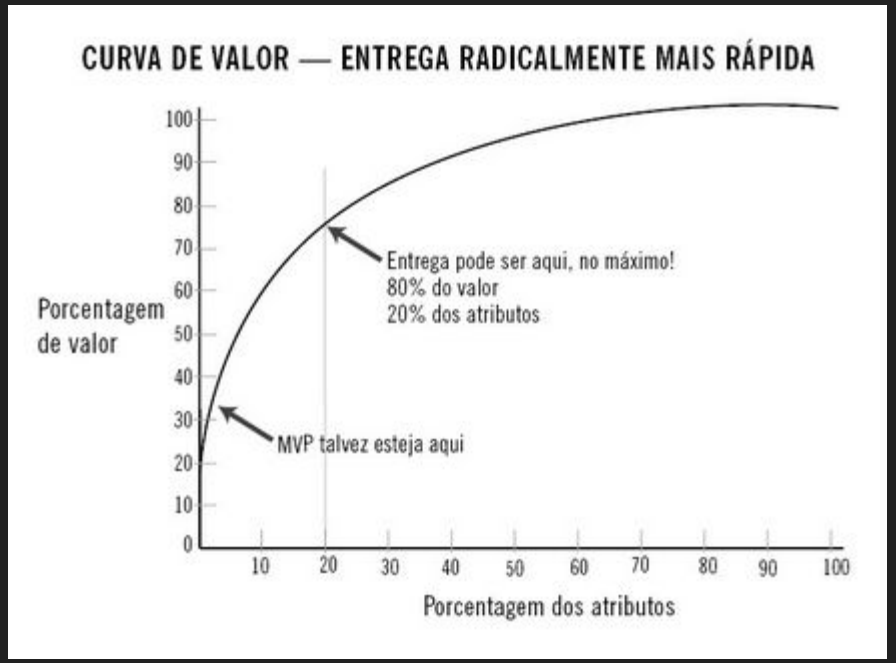
- ➔ O projeto começa o desenvolvimento depois de definir e detalhar as histórias mais importantes e definir suas prioridades
- ➔ Sem diagramas extensos, planilhas, diagramas de diagramas...
- ➔ Ao fim de cada Sprint faz-se um novo conjunto de histórias e por aí vai :)

Por que o Scrum?

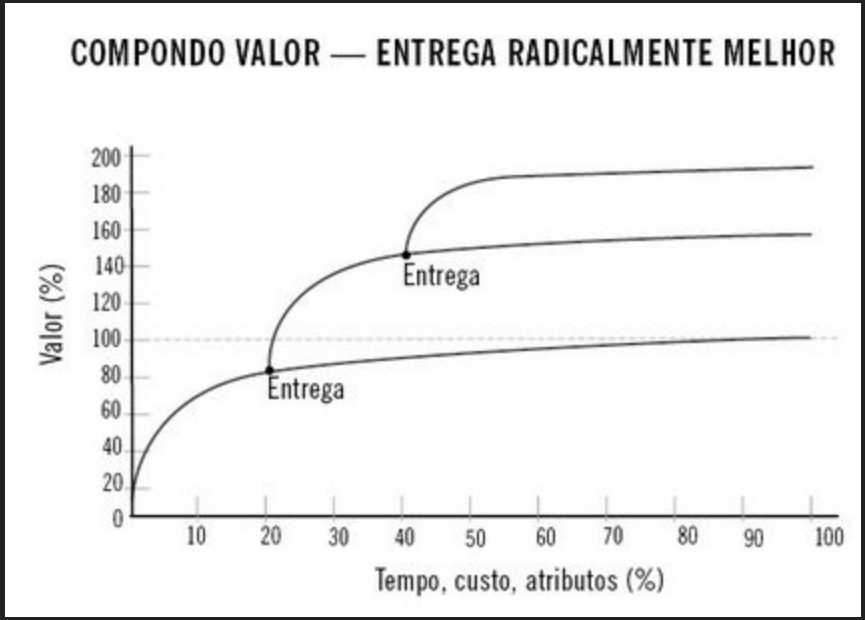
- Por fim, o Scrum permite ter algo de alto valor em pouco tempo
  - Um jogo jogável e legal :)
- Ordenando as histórias por maior valor e menor risco no backlog, e obedecendo às máximas
  - “Faça tudo certo de primeira”
  - “Fazer pela metade é igual não fazer nada”
- Ao fim de cada Sprint temos algo jogável e com valor :)

Por que o Scrum?

- Estima-se que com Scrum é possível entregar um produto com 80% do valor em 20% do tempo

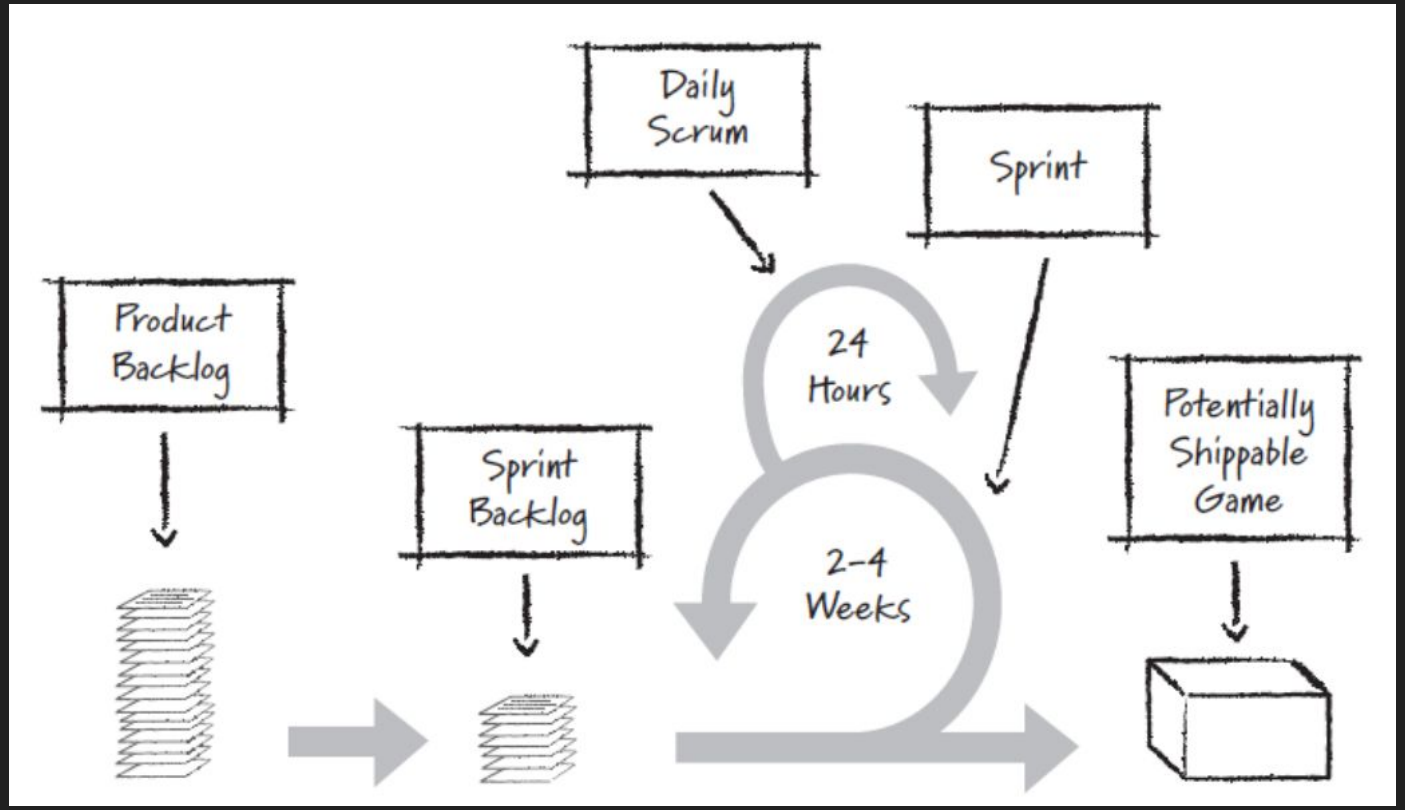


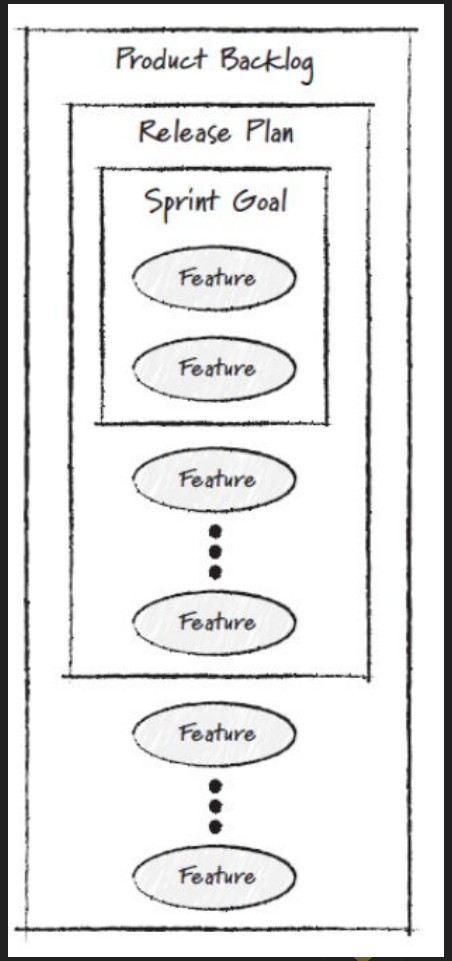
Fonte:
https://cdn-images-1.medium.com/max/800/1*b7E8lzEYhet5-e1uO03pZg.png



Fonte:
https://cdn-images-1.medium.com/max/800/1*aMup0G8w-WanhMZampaoBA.png

O cronograma do Scrum


Daily Scrum
Sprint
Product Backlog
24 Hours
Sprint Backlog
Potentially Shippable Game
2-4 Weeks


Product Backlog
Release Plan
Sprint Goal
Feature
Feature
Feature
Feature
Feature
Feature

User Stories

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

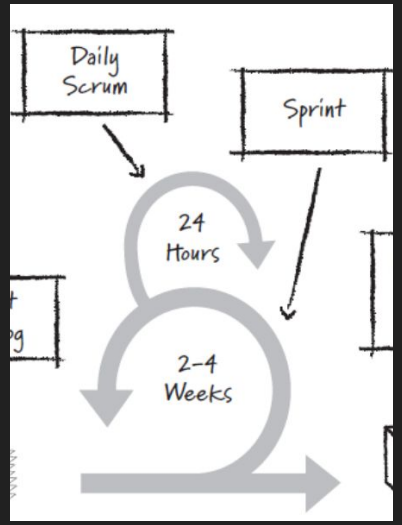
User Stories

- “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:
  - Reunir grupo completo
  - Priorizar backlog de produto
  - Montar backlog de sprint
  - Repetidos daily scrums
  - Sprint review
  - Retrospectiva



Product Backlog

➔ Conjunto de todas as user stories do produto

O jogador pode pular

Variedade de inimigos

O jogador pode atirar

Ambientação futurista

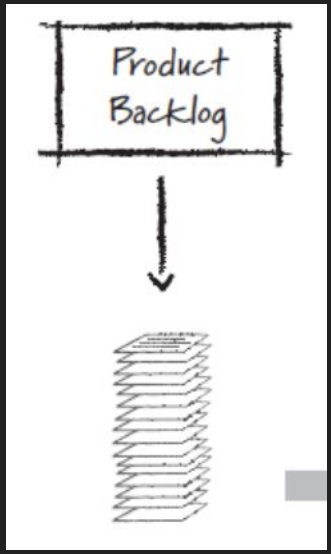
Gameplay rápida

Player aprende poderes dos chefes

O jogador pode andar

Chefes são desafiadores

Controles responsivos

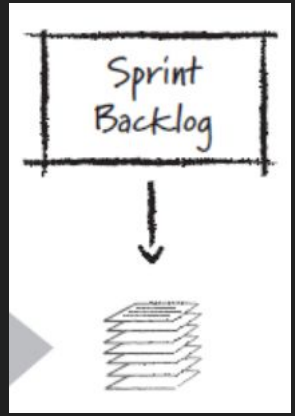


Sprint Backlog

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

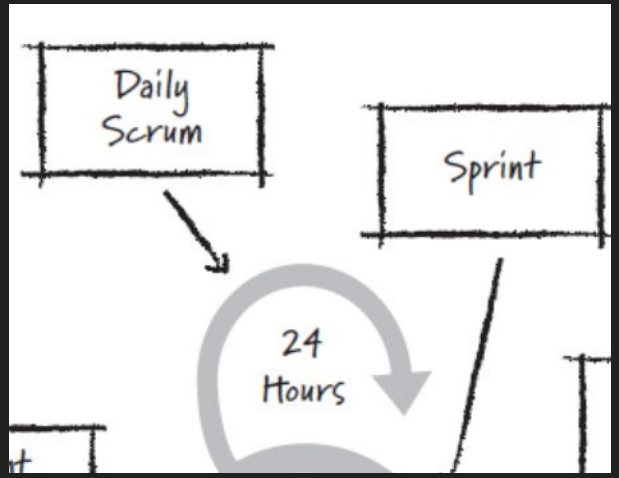
O jogador pode pular

- 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

Daily Scrum

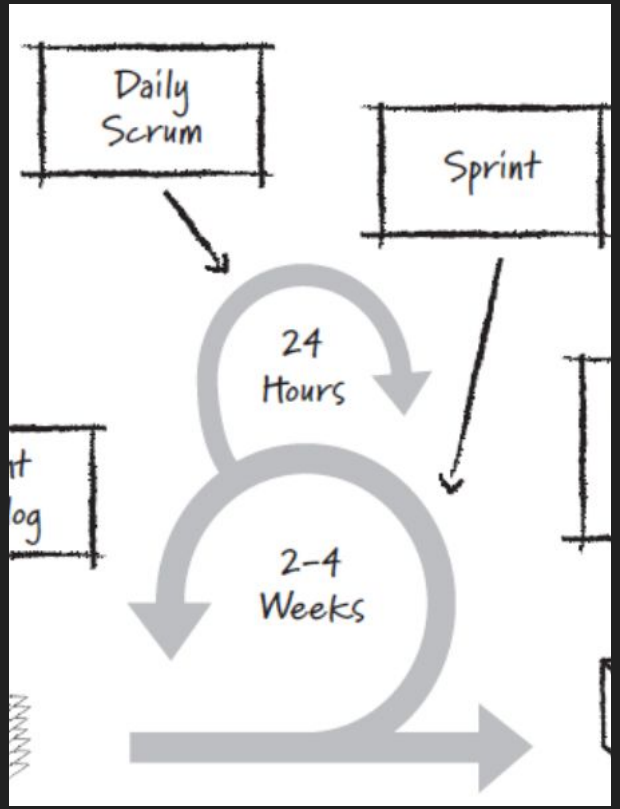


- Reunião rápida, de 5 a 10 minutos
  - "O que eu fiz desde o daily scrum anterior?"
  - "O que farei até o próximo daily scrum?"
  - "Quais são os impedimentos que estão me atrasando?"

Não é uma reunião para solução de problemas

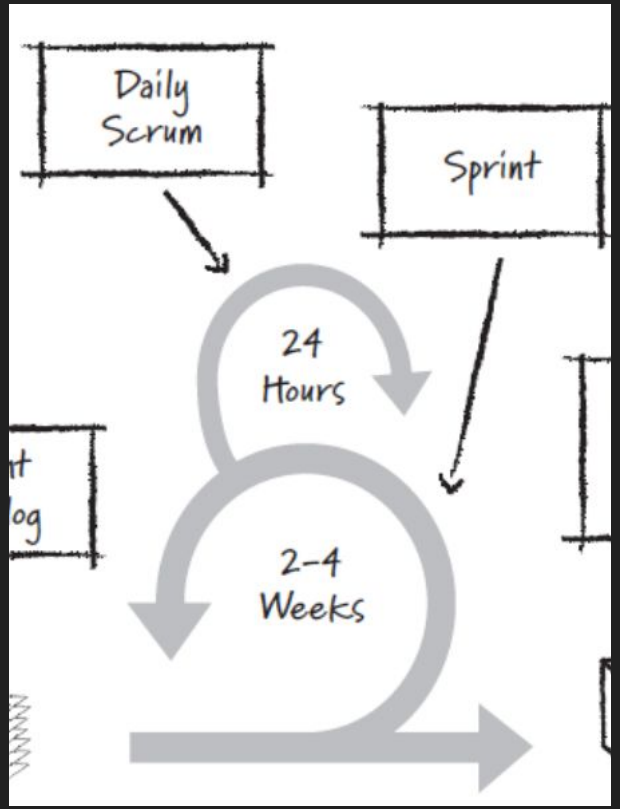
Sprint Review

- 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?"



Equipe SCRUM

Equipe SCRUM - Scrum Master

- Garantir a correta condução do SCRUM
- *Coach* da equipe SCRUM para cumprir os valores, princípios, teoria, práticas e regras
- Apoiar as atividades do Product Owner e da equipe de desenvolvimento
- Remover impedimentos para atingir os objetivos do projeto
- Fazer acontecer

Equipe SCRUM - Product Owner

- Ser o representante da organização interessada no produto
- Maximizar o valor do produto entregue pela equipe de desenvolvimento
- Definir e priorizar as funcionalidades (requisitos) do produto
- Gerenciar o Backlog do Produto
- Garantir o correto entendimento do Backlog do Produto pelos membros da equipe

Equipe SCRUM - Time de Desenvolvimento

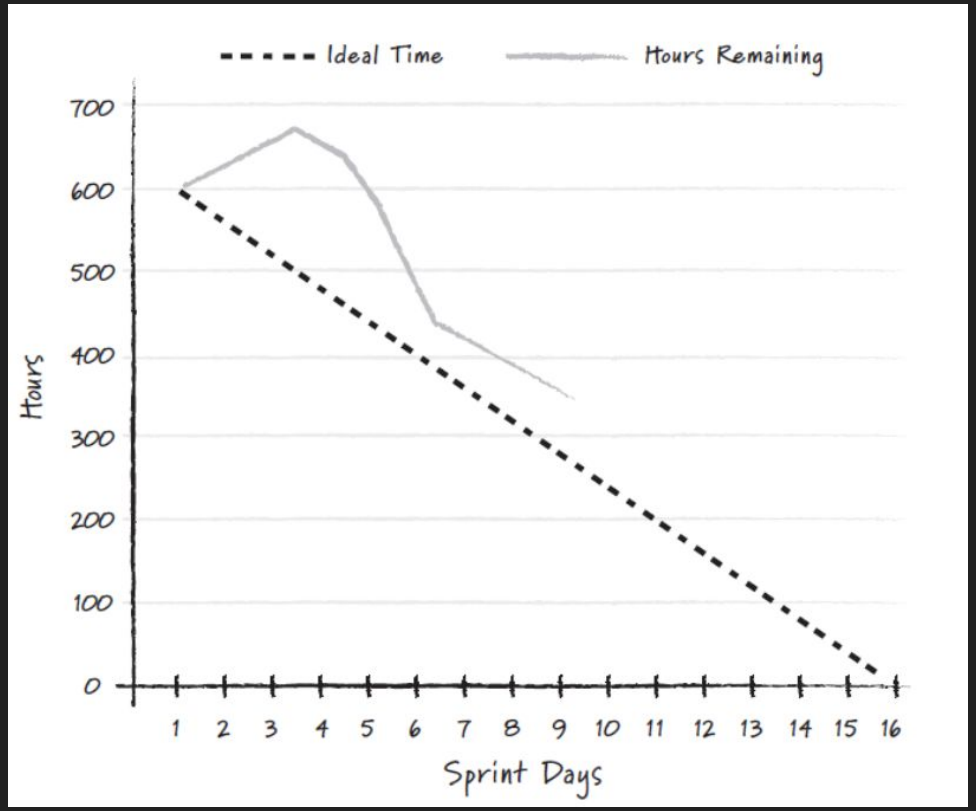
- ➔ Entregar periodicamente versões funcionais do produto
- ➔ Ser uma equipe auto-gerenciada → Organizar e gerenciar seu próprio trabalho
- ➔ Ser uma equipe multidisciplinar → juntando habilidades de programação, arte, música, design...
- ➔ Transformar o Backlog do Produto em versões do produto funcionando → Incrementos do Produto

Ferramentas para organização da equipe

- Trello
  - https://trello.com/
- Outras opções:
  - https://hacknplan.com/
  - https://www.redmine.org/
  - https://keep.google.com/

Agile Game Development with Scrum
- Clinton Keith

Burnout



Referências

CHANDLER, H. M. *Manual de Produção de Jogos Digitais*. Bookman, 2012.

GUIDE, A. *Project Management Body of Knowledge (PMBOK® GUIDE)*. In: Project Management Institute. 2001.

KEITH, Clinton. *Agile game development with Scrum*. Pearson Education, 2010.

RABIN, S. *Introdução ao desenvolvimento de games: vol. 4: a indústria de jogos: produção, marketing, comercialização e direitos autorais*. Editado por Steve Rabin, 2012.

SUTHERLAND, J. Scrum: a arte de fazer o dobro do trabalho na metade do tempo. Leya, 2016.