

# PRODUCT BACKLOG

Aula de Luiz Eduardo Guarino de Vasconcelos

# Product Backlog

## Introdução

O PO é a única pessoa responsável por gerir o Product Backlog e assegurar o valor do trabalho feito pelo Team. Este papel mantém o Product Backlog e assegura que o mesmo esteja visível a todos os envolvidos

Schwaber 2009

# Product Backlog

## Introdução

- Principal artefato do PO
- Lista priorizada de todas as funcionalidades desejadas no projeto
  - ▣ Requisitos funcionais
  - ▣ Requisitos não funcionais
  - ▣ Configuração do ambiente de desenvolvimento
- É gerado incrementalmente (grooming)
  - ▣ Começa pelo básico, o extra aparece com o tempo
- A lista é priorizada pelo PO (priorizar por maior valor)
  - ▣ A lista deve ser **ordenada pela prioridade**.
- Repriorizada no início de cada Sprint.

# Product Backlog

## Introdução

- ❑ Deve conter características que agreguem **valor de negócio** ao produto
- ❑ Novos requisitos aparecem quando o cliente vê o produto
- ❑ A arquitetura do sistema surge enquanto o projeto surge e é refatorado
- ❑ Usualmente é criado no Sprint Planning Meeting

# Product Backlog

## Qualidades desejáveis de um Product Backlog

### □ Qualidades DEEP

#### ▣ Propriamente Detalhado (D)

- Devemos achar o nível adequado de granularidade das funcionalidades do Product Backlog

#### ▣ Estimado (E)

- Equipe deve ser capaz de estimar cada item

#### ▣ Emergente (E)

- Produto está em constante mutação
- Surgem novos requisitos
- Requisitos deixam de existir
- Prioridade pode mudar

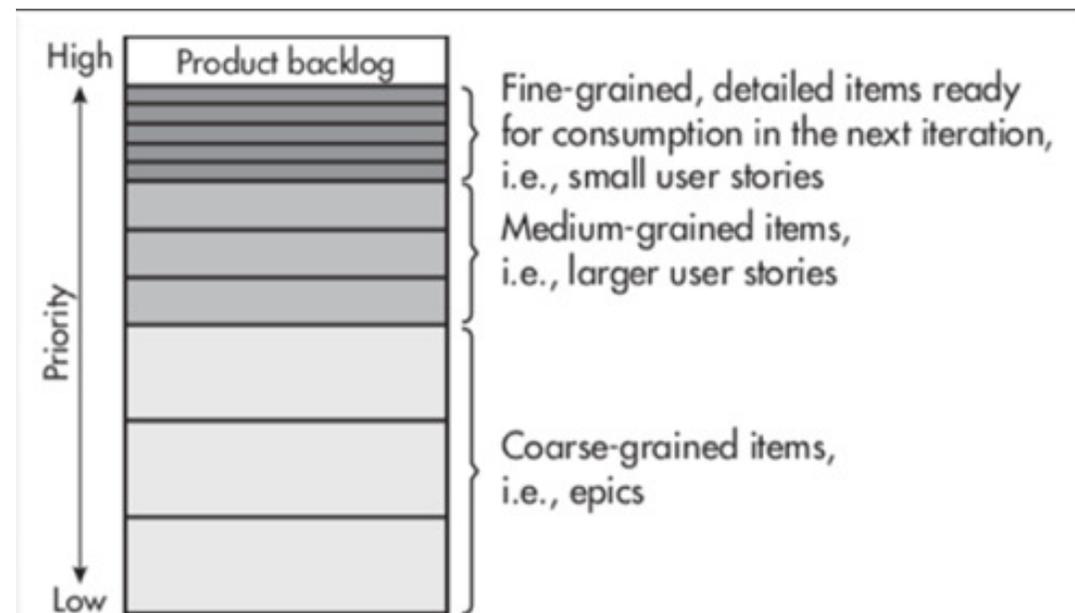
#### ▣ Priorizado (P)

# Product Backlog

## Qualidades desejáveis de um Product Backlog

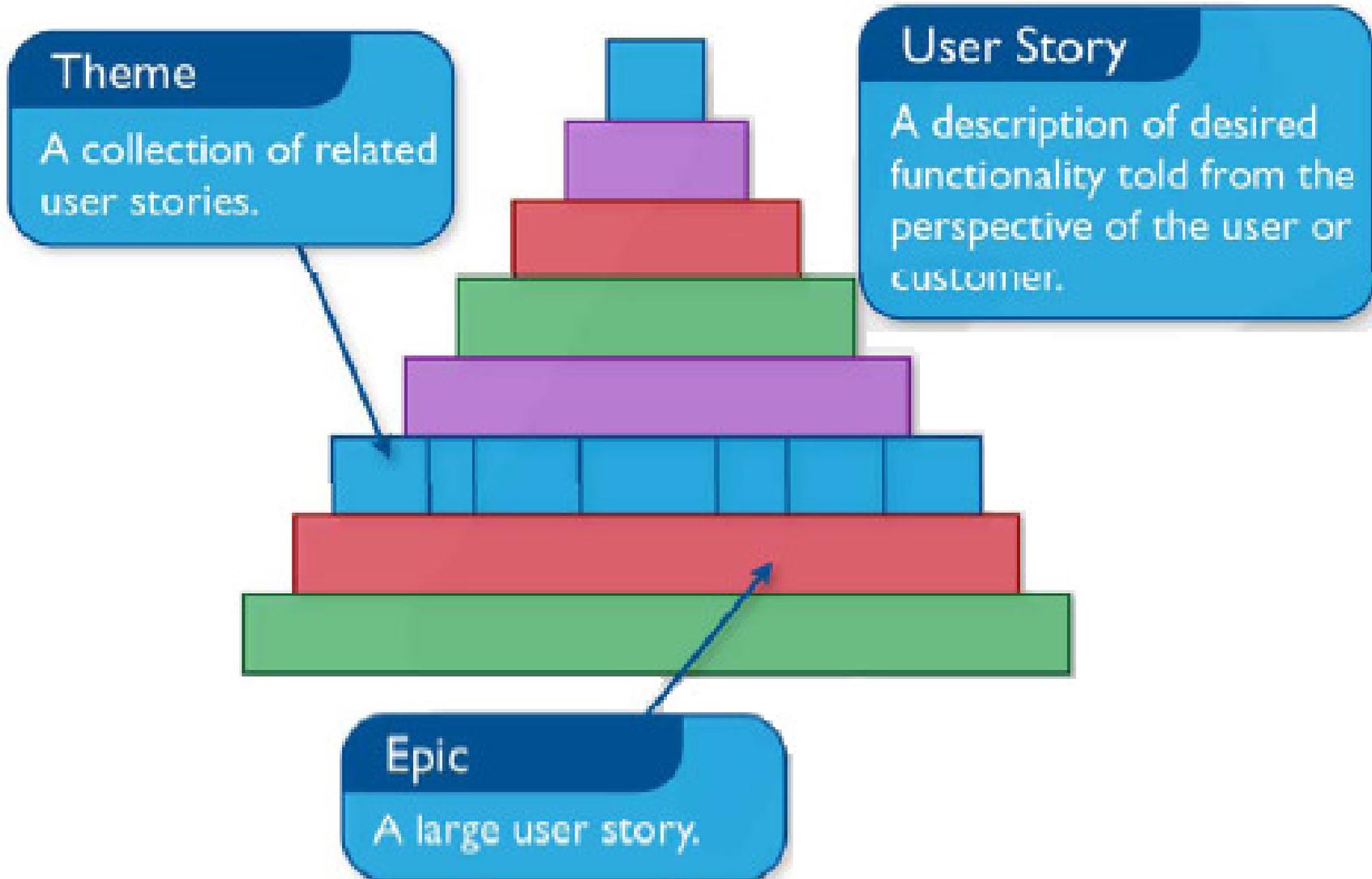
- Detalhado
  - Detalhamento depende da Prioridade
  - Maior a prioridade, mais fina a granularidade, maior a quantidade de detalhes
  - Itens na parte superior deverão ser desenvolvidos nas próximas sprints
  - Product Backlog deve ser detalhado no momento oportuno que será desenvolvido

### □ User Stories



### □ Epics

# Theme, Epic e User Story



# Product Backlog

## Qualidades desejáveis de um Product Backlog

- Estimado
  - ▣ Estimativas de granularidade grossa
  - ▣ Normalmente expressas em pontos de história (Story Points) ou dias ideais
- Saber o tamanho do item ajuda a priorizá-lo
  - ▣ Uso de horas é feito na Sprint Backlog onde haverá maior detalhamento das user stories.
- Uma vez terminado o item, ele pode ser removido do Product Backlog

# Product Backlog

## Grooming

- Precisa ter atenção e cuidado regulares
- Cuidar é um Processo Contínuo
  - ▣ Novos itens surgem
  - ▣ Itens são alterados/removidos
  - ▣ Prioridades mudam
- Os itens de maior prioridade são preparados anteriormente ao próximo Sprint Planning. Eles são quebrados e refinados.
  - ▣ Não precisa chegar com os itens prontos para a reunião, mas iniciados. Na reunião os itens deverão ser completamente detalhados.

# Product Backlog

## Grooming

- O Time estima o tamanho dos itens de **todo o Product Backlog**
  - ▣ Soma total das estimativas permite ter noção do tamanho do projeto
  - ▣ Uso do **Planning Poker** para estimar
  - ▣ Permite saber quantas sprints serão necessárias
- Scrum aloca 10% da disponibilidade do Time para atividades de gerência do Product Backlog (Schwaber, 2007)
  - ▣ Time não mais recebe requisitos. Time passa a ser coautor dos requisitos.

# Product Backlog

## Grooming

- Grooming deve acontecer em todo o ciclo de vida Scrum
  - Sprint Planning
  - Revisão da Sprint
  - Pode ser feito o detalhamento após/antes do Daily Scrum
    - Requisitos são descobertos e detalhados diariamente
- Crie uma política de grooming dentro da organização

# Product Backlog

## Grooming

- Passos para cuidar do Product Backlog
  1. Descobrir e descrever os itens
    - Time, SM, PO, stakeholders discutem os itens necessários para o produto.
    - Visão e RoadMap podem servir de ponto de partida
    - Scrum **não determina** como os itens devem ser descritos
      - User Stories (mais usado)
      - Casos de uso
      - Requisitos de usabilidade (rascunhos ou protótipos)

# Product Backlog

## Grooming

- Passos para cuidar do Product Backlog
  2. Estruturar o Product Backlog
  3. Priorizar
    - User stories no topo (maior detalhamento)
    - Themes na parte central
    - Epics no parte inferior (menor detalhamento)
- Se existirem muitos itens a serem priorizados, isso pode ser cansativo
  - Priorize pelas épicas
  - Priorize pelos temas
  - Priorize pelas user stories

# Product Backlog

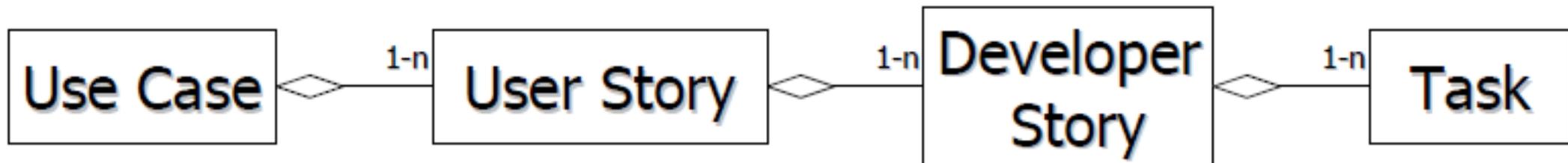
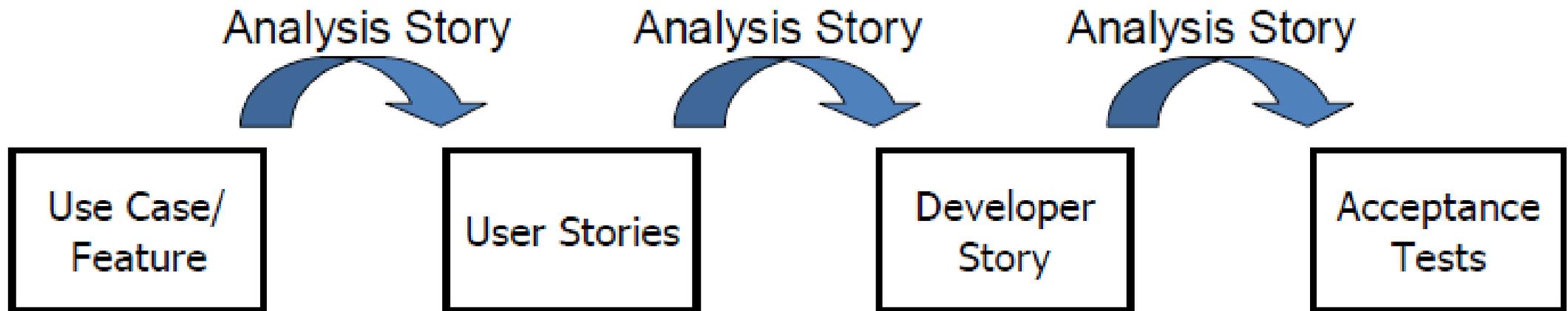
## Grooming

- Detalhamento **não deve** ser feito *a priori* para todos os requisitos (igual acontece no Cascata)
  - ▣ **Detalhe apenas os requisitos não funcionais o quanto antes**
- Foco na **funcionalidade mínima necessária e simplicidade**

# User Story

- O que deve conter?
  - ▣ Valor – requisito de um cliente
  - ▣ Custo – entendendo o que precisa para construir
  - ▣ Validação – entendimento do que está acontecendo
- Normalmente, mas nem sempre, um Caso de Uso contém diversas User Stories.

# A Transição



# User Story

## □ Uso de Cartão

- ▣ Escrever o suficiente para entender o requisito
- ▣ Palavras são imprecisas e, normalmente, precisam ser discutidas.
  - Devem ser compreensíveis
  - Encorajam o desenvolvimento iterativo
  - Projeto participativo e colaborativo

## □ Conversação

- ▣ Maioria dos requisitos são entendidos por conversação.

## □ Confirmação

- ▣ Como sabemos que o que fizemos está correto?
- ▣ Testes de aceitação

# User Story

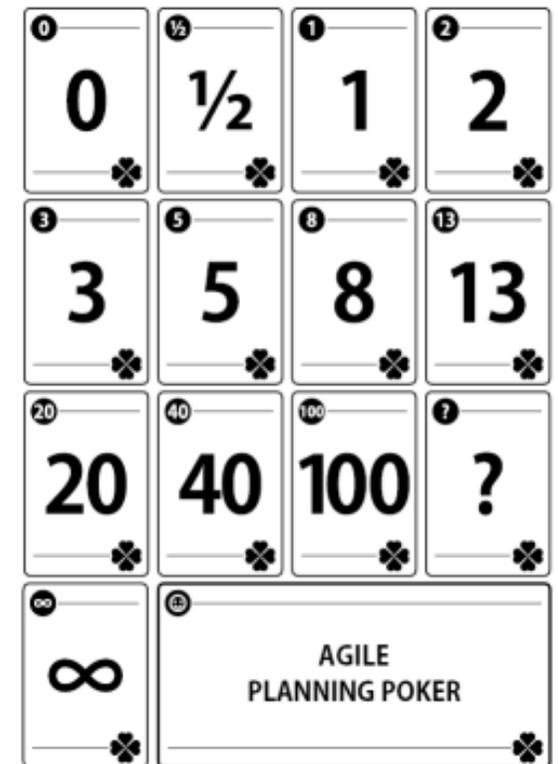
- Diretrizes
  - ▣ Como <usuário, ator>
  - ▣ Quero <funcionalidade>
  - ▣ [Para <benefício>]

# Estimativa dos Itens da Product Backlog

- O Time estima
- Devemos considerar na estimativa alguns atributos sobre a User Story
  - Esforço
  - Incerteza
  - Complexidade
- Estimar facilita a priorização
- Estimativa deve ser feita dentro da Sprint Planning e nas atividades de Grooming

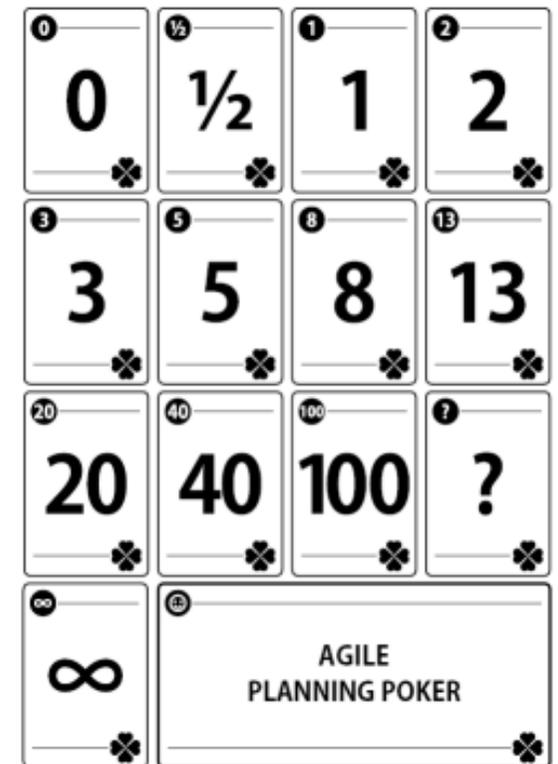
# Estimativa dos Itens da Product Backlog

- Técnica mais utilizada é **Planning Poker**
  - Estimativa **relativa** (uma user story em relação a outra)
    - Uso de **Story Points**
  - Estimativa **absoluta** (usada no Sprint Backlog)
    - **Horas**



# Estimativa dos Itens da Product Backlog

- 0 a 20 conhecemos Esforço, Incerteza e Complexidade
- 40 a 100 não conhecemos um/dois dos atributos
- “?” não conhecemos nenhum dos atributos



# Estimativa dos Itens da Product Backlog

- Funcionamento do Planning Poker
  - ▣ Todos devem ser reunidos (PO, SM, Time)
    - Time faz as estimativas
    - PO e SM podem colaborar, mas não influenciar
    - PO deve estar presente para sanar dúvidas
  - ▣ As cartas são distribuídas para todos
  - ▣ O Time determina uma User Story com Story Point 2.
    - Esta User Story será usada de comparação para as demais
    - Teremos Story Points com menor e maior valor
  - ▣ Scrum Master seleciona um item da Product Backlog
  - ▣ Todos avaliam o Esforço, Complexidade e Incerteza e atribuem uma nota ao mesmo tempo (cont.)

# Estimativa dos Itens da Product Backlog

- Funcionamento do Planning Poker (cont)
  - ▣ Em caso de consenso, passa-se para o próximo item.
  - ▣ Caso não haja consenso, aqueles que são divergentes explicam as notas
    - Normalmente, aqueles que atribuíram a menor e a maior nota são chamados a explicar (rapidamente o motivo da nota)
    - Após isso, outras rodadas são feitas até que chegue a um consenso
  - ▣ Não deve ser feita comparação de tamanho de User Story entre times distintos mesmo que seja o mesmo produto.

# Priorização

- Critérios para priorização
  - ▣ Valor
    - Item possui valor se ele é necessário para fazer do produto um sucesso
  - ▣ Conhecimento
  - ▣ Incerteza
  - ▣ Risco
    - Não existe projeto sem risco
  - ▣ Dependências (tente eliminar as dependências)
    - Podem combinar as histórias em uma história maior
    - Fatiar de forma diferente os requisitos

# Priorização

- Exemplo:
  - ▣ Como USER Quero escrever uma mensagem de texto
  - ▣ Como USER Quero escrever uma mensagem de e-mail
- São dependentes pois ambas precisam do processamento de texto
  - ▣ Pode ter outro requisito que é implementar o processamento de texto
  - ▣ As outras duas user stories seriam mais simples e independentes

# Priorização

- GUT
  - ▣ Gravidade
  - ▣ Urgência
  - ▣ Tendência
- Como aplicar
  - ▣ Listar itens;
  - ▣ Dar uma nota, de 1 a 5 para a gravidade;
  - ▣ Dar uma nota, de 1 a 5 para a urgência;
  - ▣ Dar uma nota, de 1 a 5 para a tendência;
  - ▣ Multiplicar os resultados (gravidade x urgência x tendência) de cada item;

# Priorização

## □ GUT

### ▣ Gravidade

- 5 = extremamente grave
- 4 = muito grave
- 3 = grave
- 2 = pouco grave
- 1 = sem gravidade

### ▣ Urgência

- 5 = precisa de ação imediata
- 4 = é urgente
- 3 = o mais rápido possível
- 2 = pouco urgente
- 1 = pode esperar

### ▣ Tendência (para melhor mensurar esse aspecto, sugere-se pensar "se nada for feito...")

- 5 = ...irá piorar rapidamente
- 4 = ...irá piorar em pouco tempo
- 3 = ...irá piorar
- 2 = ...irá piorar a longo prazo
- 1 = ...não irá mudar

Problema	Gravidade	Urgência	Tendência	Prioridade
Problema 1	5	5	5	125
Problema 2	4	3	1	12

# Requisitos Não Funcionais

- Devem ser identificados e implementados o quanto antes
- Podem estar mencionados na Product Backlog
  - ▣ Usabilidade, Performance, Segurança
- Incluir estes requisitos na definição de Pronto tem um preço
  - ▣ Influencia nas estimativas do Time
- O Time deve saber o que deve ser feito para entregar um item
- **Deve ter sempre um critério de aceitação**

# Requisitos Não Funcionais

- Exemplo:
  - ▣ Restrição de Performance
    - O sistema deve responder uma solicitação em menos de 1 segundo
  - ▣ Critério de aceitação
    - 10 mil transações de leitura / escrita ao mesmo tempo
    - Cada transação tem um tamanho de no máximo 500 KB

# Artefatos

- Product Backlog é o principal artefato mas não é o único
- Podem ser feitos outros artefatos
  - ▣ Resumo dos papéis dos usuários
  - ▣ Workflows
  - ▣ Diagramas
  - ▣ Planilha de cálculos
  - ▣ Rascunhos/protótipos UI
- Estes artefatos não substituem o Product Backlog

# Product Backlog

## Erros Comuns

- ❑ Apesar de simples, pode ser difícil de ser usado da forma correta
- ❑ Especificação de requisitos disfarçados de backlog
  - ▣ Garanta a comunicação correta entre Todos
- ❑ Lista de presentes para o Papai Noel (quer tudo)
  - ▣ Atenda o Produto Mínimo, descarte o Resto.
- ❑ Empurrar a definição de requisitos para o Time
  - ▣ Time deve definir junto. Foco sempre em “NÓS”.
- ❑ Negligenciar o Grooming
- ❑ Fornecer vários backlogs para o mesmo Time