

# Aula 10 - Priorização do Backlog

**Disciplina:** Gestão de Projetos Tecnológicos

**Prof. Me. João Paulo Biazotto**

# Objetivos da Aula

- Entender o papel da priorização no gerenciamento ágil.
- Conhecer 3 técnicas de priorização: MoSCoW, Matriz Valor x Esforço e WSJF.
- Aplicar exemplos práticos de priorização em backlog fictício.
- Refletir sobre boas práticas e erros a evitar.

# O que é Backlog

- Lista **ordenada** de tudo que pode ser desenvolvido em um produto.
- É **dinâmica**: muda conforme aprendizados, feedback e contexto de mercado.
- É responsabilidade do **Product Owner** manter o backlog visível, transparente e priorizado.

# Por que Priorizar?

- Recursos são **limitados**: tempo, orçamento, equipe.
- Nem tudo pode ser feito ao **mesmo tempo**.
- A priorização garante que o time trabalhe primeiro no que **traz maior valor**.
- Ajuda a **alinhar expectativas** entre stakeholders.

# Por que Priorizar?

- Recursos são **limitados**: tempo, orçamento, equipe.
- Nem tudo pode ser feito ao **mesmo tempo**.
- A priorização garante que o time trabalhe primeiro no que **traz maior valor**.
- Ajuda a **alinhar expectativas** entre stakeholders.

# Consequências da Má Priorização

- Entrega de funcionalidades **pouco relevantes**.
- **Retrabalho** por não atender as necessidades principais.
- Clientes **insatisfeitos**.
- Atraso no **time-to-market**.
- **Desmotivação** da equipe.

# Mas como priorizar?

Para evitar esses problemas, existem diferentes técnicas de priorização. Vamos conhecer três delas e ver exemplos práticos:

- MoSCoW
- Matriz Valor x Esforço
- WSJF

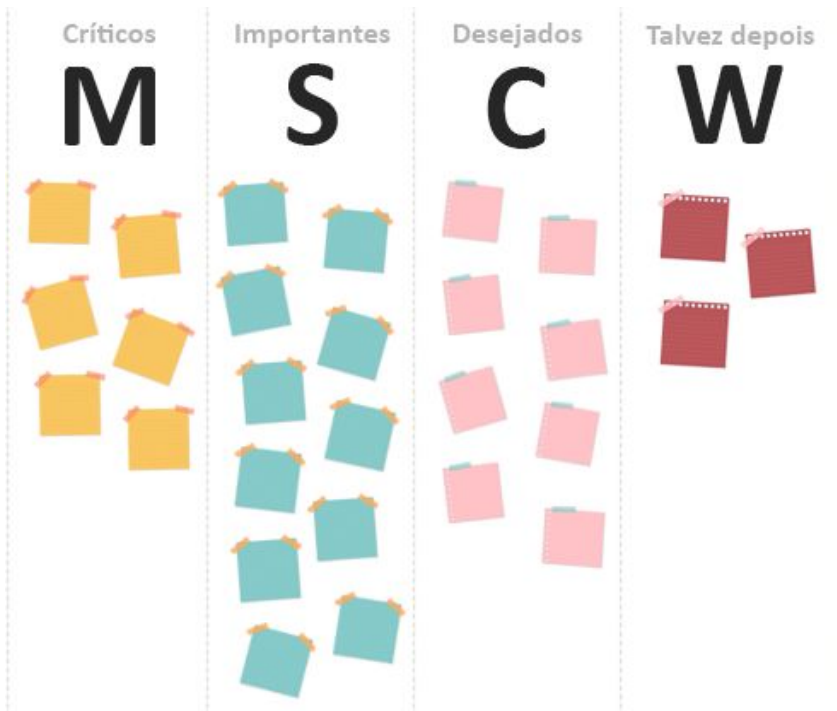


# Técnica 1 – MoSCoW

- Método criado para ajudar a priorizar requisitos.
- Divide itens em 4 categorias.
- Fácil de aplicar em equipes pequenas e médias.

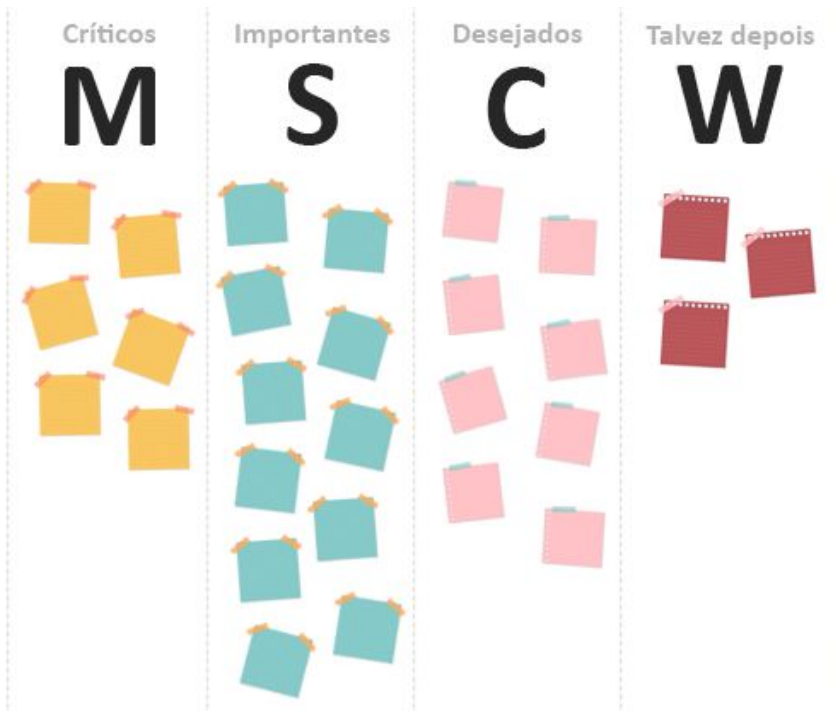


# Categorias do MoSCoW



- **(M)ust Have (Críticos):** essenciais, sem eles o produto não funciona.
- **(S)hould Have (Importantes):** importantes, mas não essenciais no momento.
- **(C)ould Have (Desejados):** desejáveis, trazem valor adicional.
- **(W)on't Have (Talvez depois):** não serão feitos agora.

# Passo a Passo para aplicação



1. Reunir o backlog.
2. Classificar cada item em uma das 4 categorias.
3. Revisar em conjunto com os stakeholders.
4. Validar e refinar conforme necessário.

# Exemplo de Aplicação: App de Delivery

- Pagamento com cartão.
- Avaliação de restaurantes.
- Filtro por preço.
- Integração com cupons.
- Rastreamento do pedido.

# Exemplo de Aplicação: App de Delivery

- Pagamento com cartão. (M)
- Avaliação de restaurantes.
- Filtro por preço.
- Integração com cupons.
- Rastreamento do pedido. (M)

# Exemplo de Aplicação: App de Delivery

- Pagamento com cartão. (M)
- Avaliação de restaurantes.
- Filtro por preço. (S)
- Integração com cupons.
- Rastreamento do pedido. (M)

# Exemplo de Aplicação: App de Delivery

- Pagamento com cartão. (M)
- Avaliação de restaurantes. (C)
- Filtro por preço. (S)
- Integração com cupons.
- Rastreamento do pedido. (M)

# Exemplo de Aplicação: App de Delivery

- Pagamento com cartão. (M)
- Avaliação de restaurantes. (C)
- Filtro por preço. (S)
- Integração com cupons. (W)
- Rastreamento do pedido. (M)

# Benefícios do MoSCoW

- Simples e rápido de aplicar.
- Ajuda a alinhar stakeholders.
- Torna clara a diferença entre o que é essencial e o que é opcional.



# Limitações do MoSCoW

- Pode ser subjetivo se não houver critérios claros.
- Nem sempre considera esforço ou custo.
- Risco de inflar o “Must Have” se não houver disciplina.

# Quando utilizar o MoSCoW

- Útil quando o foco é escolher o **mínimo** viável.
- Ajuda a identificar o **MVP** (Produto Mínimo Viável).
- Melhor aplicada em contextos com **muitos requisitos iniciais**.

## Técnica 2 - Matriz Valor x Esforço

- Técnica visual que cruza:
  - Valor gerado pelo item.
  - Esforço necessário para implementá-lo.

# Passo a Passo para construir a Matriz



1. Listar itens do backlog.
2. Atribuir notas de valor e esforço.
3. Posicionar na matriz.
4. Definir ordem de priorização com base nos quadrantes.

# Exemplo Prático: Plataforma de Streaming

- Modo offline.
- Recomendação personalizada.
- Tema escuro.
- Melhorar performance de busca.
- Integração com redes sociais.

# Exemplo Prático: Plataforma de Streaming

- Modo offline.
- Recomendação personalizada.
- Tema escuro. (Quick Win)
- Melhorar performance de busca. (Quick win)
- Integração com redes sociais.

# Exemplo Prático: Plataforma de Streaming

- Modo offline. (Major Project)
- Recomendação personalizada. (Major Project)
- Tema escuro. (Quick Win)
- Melhorar performance de busca. (Quick win)
- Integração com redes sociais.

# Exemplo Prático: Plataforma de Streaming

- Modo offline. (Major Project)
- Recomendação personalizada. (Major Project)
- Tema escuro. (Quick Win)
- Melhorar performance de busca. (Quick win)
- Integração com redes sociais. (Baixa prioridade)



# Benefícios da Matriz Valor x Esforço

- Facilita tomada de decisão visualmente.
- Ajuda a identificar vitórias rápidas.
- Considera tanto valor quanto custo.

# Limitações da Matriz Valor x Esforço

- Avaliação de “valor” pode ser subjetiva.
- Esforço muitas vezes é apenas estimado.
- Pode exigir discussões demoradas.

# Quando utilizar a Matriz Valor x Esforço

- Excelente para negociar prioridades com stakeholders.
- Dá clareza sobre o que entregar primeiro quando há recursos limitados.

# Quando utilizar a Matriz Valor x Esforço

- Excelente para negociar prioridades com stakeholders.
- Dá clareza sobre o que entregar primeiro quando há recursos limitados.

## Técnica 3 - WSJF

- Técnica usada no SAgile (Scaled Agile Framework).
- O objetivo é **maximizar** o **valor** entregue mais rapidamente, escolhendo **primeiro** os itens que trazem **mais impacto em menos tempo**.
- Prioriza com base em **custo de atraso** dividido pelo **tamanho do trabalho**.

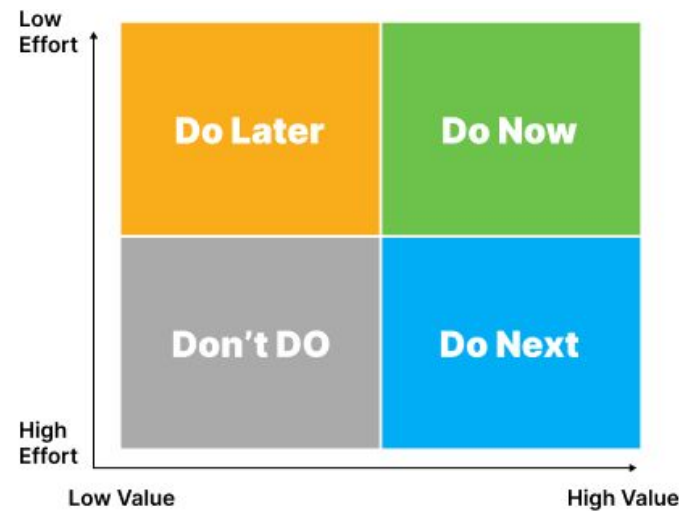
# Custo do Atraso (CoD)

- O **Custo do Atraso** mede quanto valor a organização **deixa de ganhar ou perde** por não entregar uma funcionalidade o quanto antes.
  - *Valor de Negócio*: O benefício que aquela entrega gera para clientes e para o negócio.
  - *Urgência/Time Criticality*: Quão urgente é entregar agora (por exemplo, janela de mercado, regulamentação, eventos sazonais)
  - *Redução de Risco/Oportunidade*: Se a entrega ajuda a reduzir riscos futuros ou aproveitar oportunidades importantes.

## Esforço (Job Size)

- O esforço, tempo ou complexidade necessários para concluir a tarefa.
- Pode ser medido em:
  - *Pontos de história (estimativa ágil).*
  - *Pontos de função*
  - *Dias/horas de trabalho.*
- Qualquer métrica que reflita o custo de implementação.

# Critérios do WSJF



WSJF =

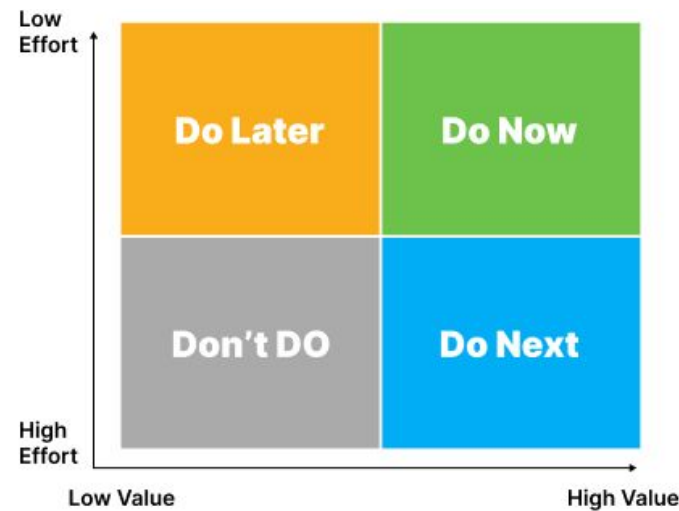
Cost of Delay

Job Duration  
(Job Size)

- **Valor de Negócio:** impacto para clientes/empresa.
- **Urgência (time criticality):** necessidade imediata.
- **Redução de Risco/Oportunidade:** evitar problemas ou aproveitar oportunidades.
- **Esforço:** tempo ou complexidade da implementação.



# Critérios do WSJF



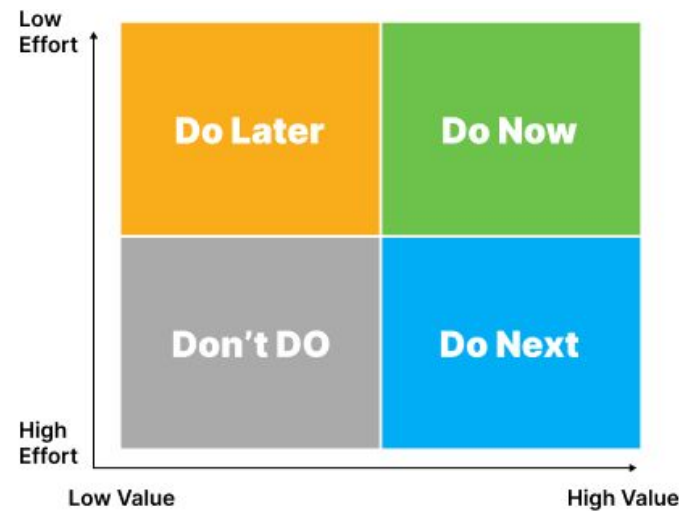
WSJF =

Cost of Delay

Job Duration  
(Job Size)

- Tarefas com alto CoD e baixo esforço terão maior WSJF → **devem ser feitos primeiro.**
- Tarefas com alto CoD mas grandes ainda podem ser importantes, mas podem ser **quebrados** em pedaços menores.

# Passo a passo para aplicar do WSJF



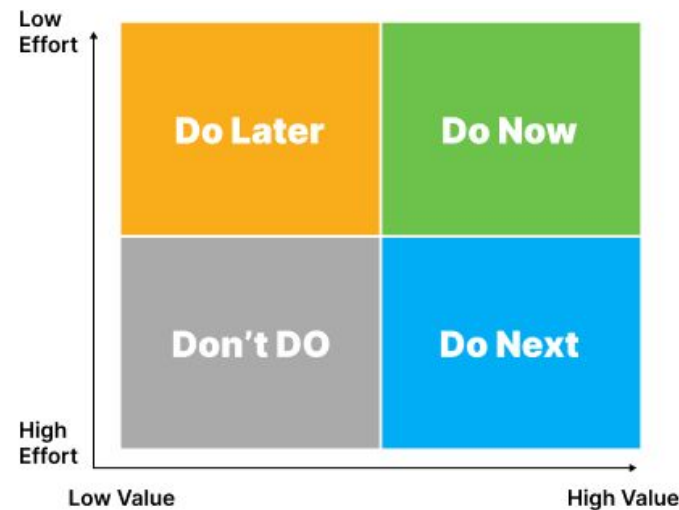
WSJF =

Cost of Delay

Job Duration  
(Job Size)

1. Definir critérios de avaliação.
2. Atribuir notas de 1 a 10 para cada critério.
3. Calcular WSJF dividindo pelo esforço.
4. Ordenar backlog pelo maior WSJF.

# Passo a passo para aplicar do WSJF



WSJF =

Cost of Delay

Job Duration  
(Job Size)

1. Definir critérios de avaliação.
2. Atribuir notas de 1 a 10 para cada critério.
3. Calcular WSJF dividindo pelo esforço.
4. Ordenar backlog pelo maior WSJF.

# Exemplo Prático: Fintech

- Cartão virtual.
- Notificação de gastos em tempo real.
- Investimento automático.
- Chatbot de atendimento.

## Exemplo Prático: Fintech

Item	Valor	Urgência	Risco	Esforço	WSJF
Cartão virtual	8	9	8	5	5.00
Notificação	9	8	7	3	8.0
Investimento automático	2	6	9	8	2.125
Chatbot	7	5	2	4	3.5

# Benefícios do WSJF

- Mais objetivo: considera múltiplos fatores.
- Permite cálculo comparativo entre itens.
- Ajuda em ambientes complexos e de grande escala.

# Limitações do WSJF

- Requer mais tempo de análise.
- Precisa de disciplina na atribuição de valores.
- Pode ser difícil em equipes inexperientes.

## Quando usar WSJF?

- Mais robusto em ambientes corporativos complexos.
- Útil em empresas que usam SAFe ou grandes portfólios.



# Quando utilizar cada técnica?

- Times pequenos/startups → MoSCoW.
- Negociação visual com stakeholders → Matriz.
- Grandes empresas/SAFe → WSJF.

# Boas práticas na priorização do backlog

- Definir critérios claros antes da priorização.
- Reavaliar backlog periodicamente.
- Envolver stakeholders.
- Focar no valor entregue ao cliente

# Erros Comuns na priorização do backlog

- Colocar tudo como “prioridade máxima”.
- Ignorar esforço/custo.
- Decidir sem critérios.
- Não revisar prioridades com frequência.

# DÚVIDAS?