**Pesquisa**

**Renan Gabriel Rodrigues Mendonça**

**Projetos**

Metodologias:

* Modelos Clássico

- Cascata (ou waterfall): O Modelo Cascata são etapas sequenciais e organizada para o processo de desenvolvimento de software, ele é utilizado principalmente quando os requisitos do projeto são bem compreendidos, geralmente ele é utilizado em sistemas existentes e serve para realizar adaptações e melhorias. Ele só avança uma etapa se a anterior for concluída. Primeiro você planeja tudo, depois analisa, projeta, implementa, testa e mantém

Onde o modelo cascata é usado

Exemplos: Software corporativos

**Vantagens:**

**-** Simplicidade e clareza

**-**Planejamento detalhado

**-**Entrega Completa e Organizada

**-** Controle de Progresso

**-** Documentação Robusta

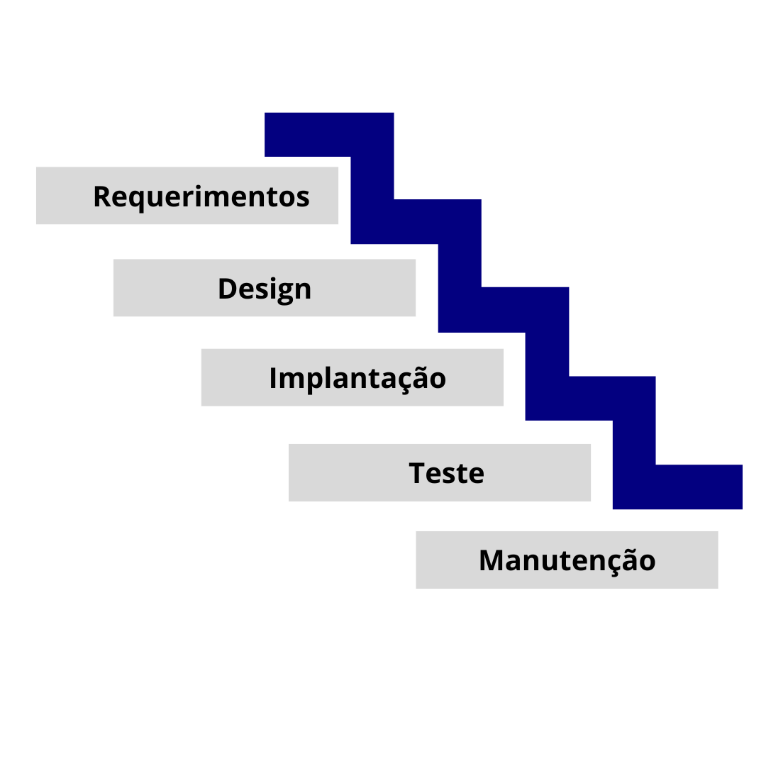
**Desvantagens:**

- Pouca flexibilidade na correção de erros

- Entrega Demorada

- Depende dos requisitos bem definidos

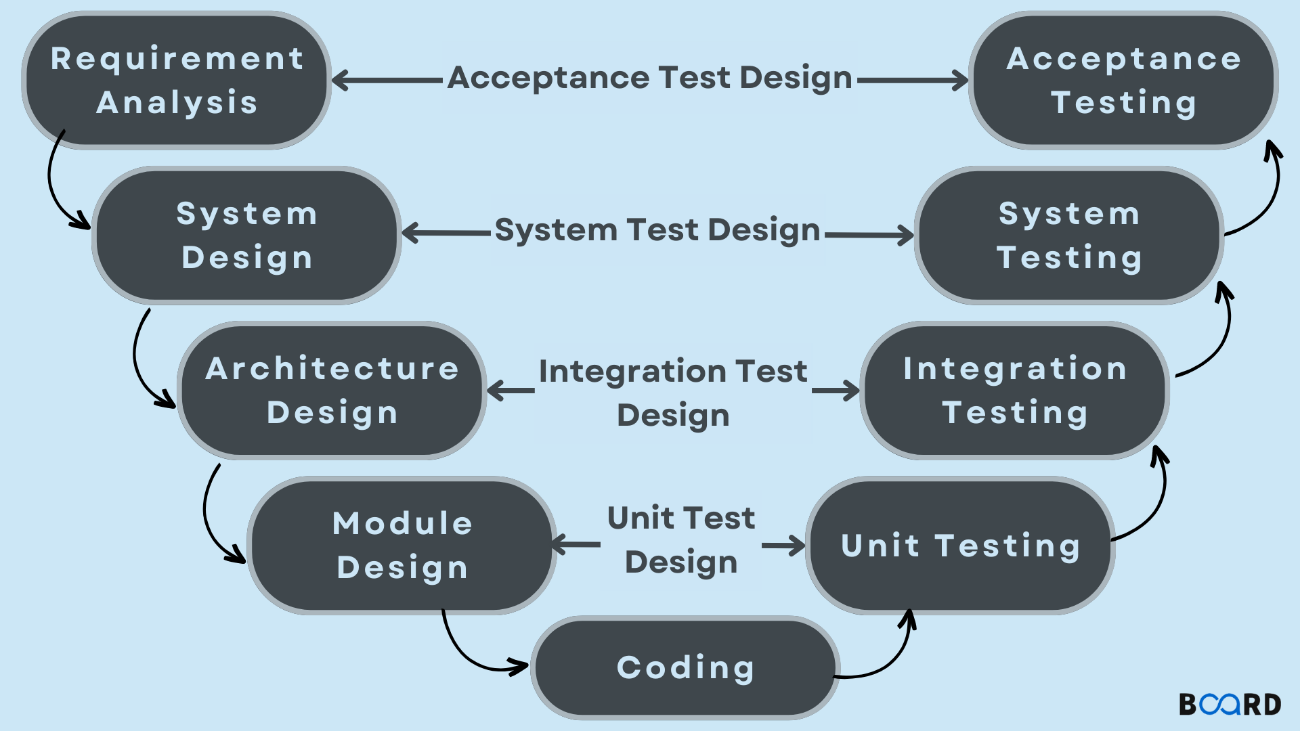
- alto risco para projetos que possuem tecnologia nova



- V-Model: Essa metodologia de desenvolvimento é uma variante do modelo cascata, seu objetivo é garantir que cada fase de desenvolvimento tenha um teste correspondente, reduzindo possíveis falhas no sistema futuramente. Ele é muito utilizado para sistema que exigem alta confiabilidade.

Onde o modelo v é usado

Exemplo: indústria automotiva, setor aeroespacial ,Área da saúde e segurança e defesa



Vantagens:

- Validação e verificação desde o início

- Ótimo em projetos críticos e de alta confiabilidade

Onde o modelo v é usado: indústria automotiva, setor aeroespacial ,Área da saúde e segurança e defesa.

-Clareza e organização

- Documentação bem completa

- menor possibilidade de retrabalho

Desvantagens:

- Pouca flexibilidade para mudanças

- Alto custo inicial

- Não é ideal para projetos com tecnologias avançada

- Cliente consegue ver o resultado só no final

- requer uma equipe experiente

Por causa dos testes, a equipe precisa ter um bom conhecimento dos requisitos

Espiral: modelo espiral é uma abordagem iterativa e incremental que combina elementos de desenvolvimento em cascata com prototipação. Consiste em ciclos de atividades, onde cada ciclo representa uma iteração. Uma metodologia desenvolvida por Barry Boehm em 1986. Ideal para projetos complexos e de grande porte, onde os requisitos podem evoluir ao longo do tempo. Cada ciclo tem quatro principais fases:

**Determinação de objetivos:** definição dos objetivos, funcionalidades e restrições para a iteração atual.

Ex: A primeira versão precisa ter uma tela de login básica.

**Análise de riscos:** Identificar possíveis riscos técnicos, de prazo, custos ou requisitos mal definidos.

EX: Se uma determinada funcionalidade seja complexa, o time pode decidir prototipar para reduzir possíveis erros.

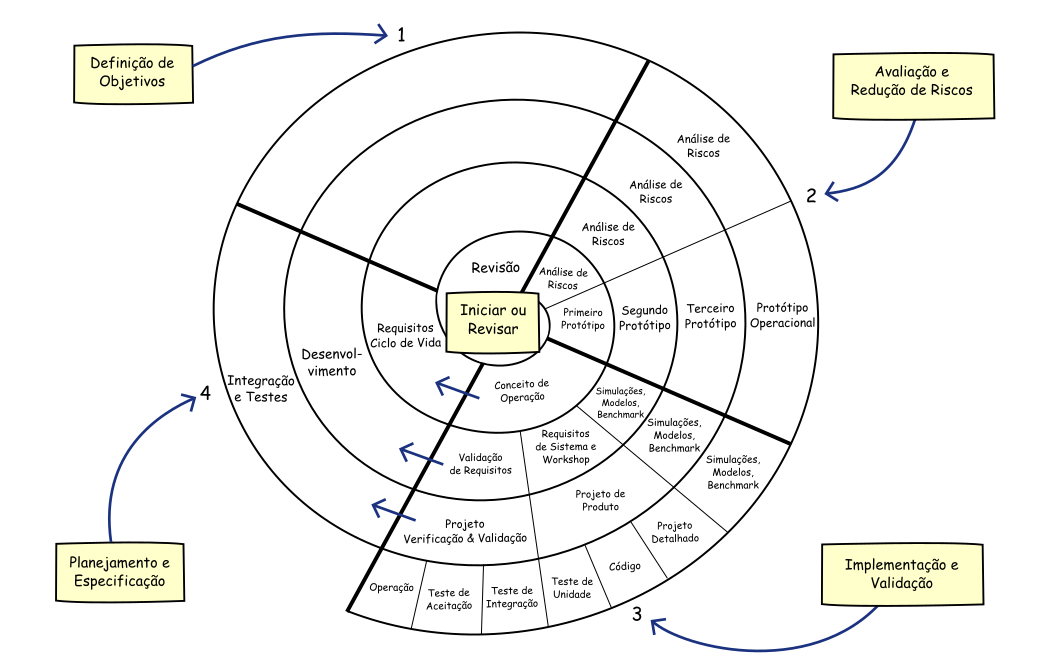
**Desenvolvimento e teste:** codificação e testes do produto baseado nos objetivos traçados.

EX: desenvolver um protótipo funcional e testar se ele atende os requisitos especificados.

**Planejamento**: Revisar o que foi feito e planejar a próxima iteração, diminuindo a probabilidade de falhas.

EX: Após o feedback do cliente, a equipe pode adicionar mais funcionalidades ou realizar alterações das funcionalidades existentes.

Essas fases são executadas de forma iterativa, permitindo o refinamento e aprimoramento do software ao longo do processo.



Vantagens:

- Analise de risco desde do início, consequentemente diminuindo a possibilidade de erros.

- Flexibilidade nas adaptações durante o processo de desenvolvimento, podendo alterar os requisitos no projeto

- Entrega de protótipos funcional a cada ciclo

-Feedbacks constantes de clientes a cada entrega

Desvantagens:

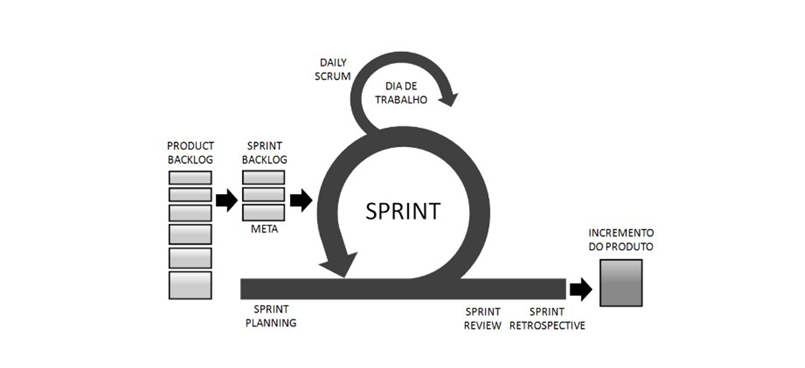
- Necessidade de recurso e tempo adicional para analise de cada risco detalhado em cada ciclo

- Complexidade em projetos de grande escala, consequentemente exigindo uma gestão eficiente e coordenada

- Possibilidade de escopo indefinido caso não seja gerenciado corretamente, consequentemente causando problemas de cronograma e orçamento

* Modelos Ágeis

-Scrum: é uma metodologia ágil baseada em sprints, os ciclos de produção de um projeto, garantindo uma revisão e aperfeiçoamento constante para que seja alcançado um ótimo resultado é o trabalho colaborativo é essencial dentro desse Modelo e esse modelo é um framework que gestores possam acompanhar um desenvolvimento desse projeto consequentemente a entrega do trabalho tende a ser melhor.



Ex: Desenvolvimento de hardware e Projetos de pesquisa científica

Vantagens:

- Maior flexibilidade

- Entrega de valor incremental, conforme o projeto vai evoluindo em cada ciclo, ele é entregue ao cliente

- Trabalho em equipe e comunicação

- Melhoria Contínua

- Visibilidade e transparência

Desvantagens:

- Possível dificuldade na definição de prazos

- Requer uma equipe altamente qualificada e autogerenciável

- Necessidade de colaboração intensa

- Riscos de sobrecarga de trabalho

- Requer um alto compromentimento total da equipe

-XP: Extreme Programming (XP) É uma metodologia Ágil que visa desenvolver sistemas de alta qualidade por meio de ciclos curtos de desenvolvimento, forte colaboração com o cliente e entregas frequentes. Ela enfatiza testes contínuos, adaptações rápidas ás mudanças do mercado e iterações curtas para garantir prazo reduzido e maior satisfação do cliente.

Exemplos: AWS (Amazon web services)



**Códigos > Testes > Ouvir o usuário > Executar o Design**

Vantagens:

- Colaboração de clientes

- Entrega rápida

- Adaptabilidade

- feedbacks

Desvantagens:

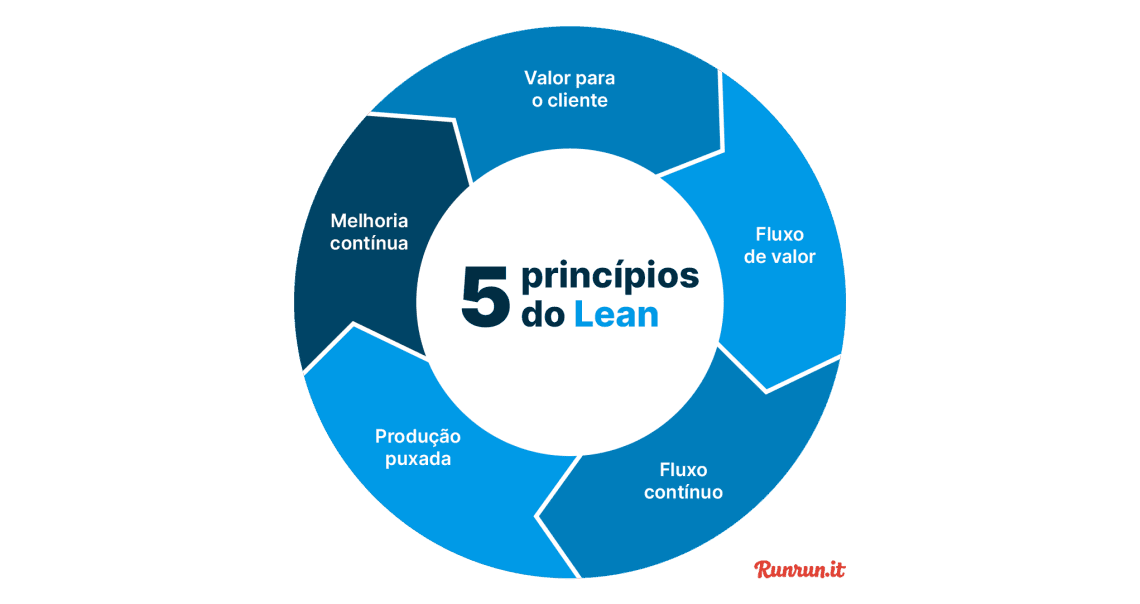
- Resistência à mudanças

- Necessidade de uma comunicação eficiente

- Foco extremamente excessivo na entrega de curto prazo

- Equipe experiente e autogerenciável

-Lean: é um modelo de gestão de processos que ao eliminar desperdícios, otimizar custos e implementar a melhoria contínua, visa tornar os fluxos de trabalhos mais eficientes. Simultaneamente, agrega o máximo de valor nas entregas com intuito de agradar os clientes



Exemplo: Ford e Toyota utilizam essa metodologia

Vantagens:

-Comunicação transparente

- investimento em capacitação para a equipe

-Equilíbrio entre custos, qualidade e satisfação do cliente

- Acompanhamento constante

Desvantagens:

- Resistência à mudança

- Necessidade de treinamento

- Foco excessivo na redução de custos

-Kanban: esse modelo é gestão visual que ajuda organizar diversas tarefas, esse tipo de modelo busca organizar, aumentar a eficiência e reduzir possíveis desperdícios. O kanban utiliza quadros visuais para acompanhar tarefas em diferentes estágios de um fluxo de trabalho de um determinado projeto.

Vantagens:

-Reduz desperdícios

- aumenta a produtividade

- Aumenta a transparência do trabalho

- Pode ser usado em várias áreas

- Identificação de tarefas que não avançam, atraso em uma etapa específica, sobrecarga de tarefas e etc.

Desvantagens:

- Falta de prazo fixos

- Dificuldade de lidar com demandas imprevisíveis

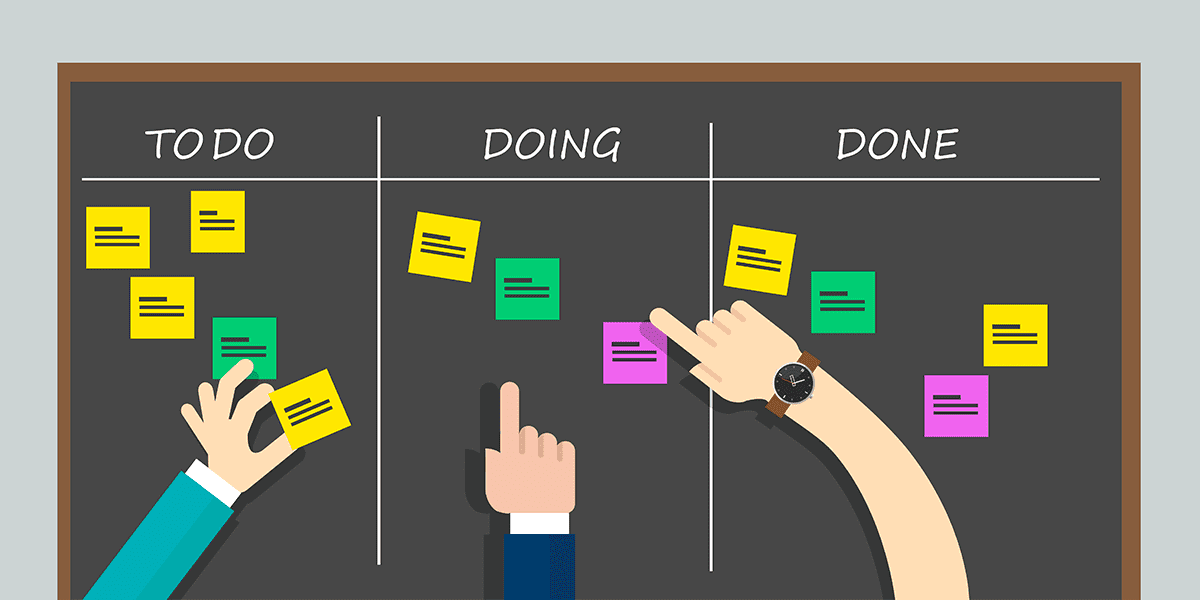
- Possível sobrecarga de trabalho

-Ineficiência para tarefas Complexas e interdependentes

-Dependência de colaboração Eficaz

-Complexidade e Confusão na implementação

-Falta de estrutura para projetos Complexos



Exemplos de uso: desenvolvimento de software e gestão de projetos

-Crystal: é um modelo que enfatiza a importância da adaptação às necessidades específicas de cada projeto, esse modelo reconhece que **não existe um processo único que funcione para todos os projetos**. Ele se baseia em princípios e valores ao invés de processos rígidos.

Esse modelo classifica os projetos por cores, baseando-se em cores através do tamanho da equipe e no nível de risco do software.

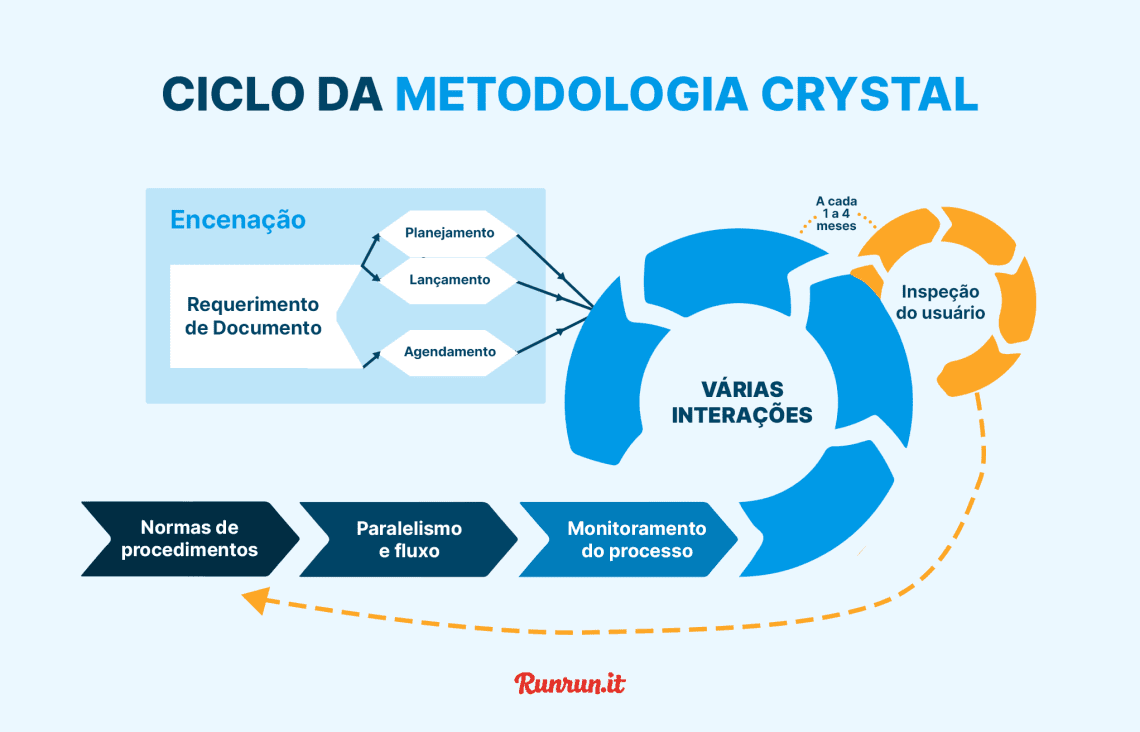
Variantes do modelo crystal

**Crystal Clear** – Para equipes pequenas (até 6 pessoas). Pouca formalidade, muita comunicação direta.

**Crystal Yellow** – Para equipes de 7 a 20 pessoas. Exige um pouco mais de estruturação.

**Crystal Orange** – Para equipes de 21 a 40 pessoas. Já precisa de mais organização e algumas documentações.

**Crystal Red** – Para grandes equipes (acima de 40 pessoas) e projetos críticos.



Vantagens:

- Adaptação ao projeto

- Foco em pessoas (prioriza a comunicação e colaboração da equipe)

- Entregas frequentes

- Melhoria Contínua

- Evita documentação excessiva

Desvantagens:

- Falta de estrutura definida

- Dependência da equipe

- Dificuldades em grandes projetos

- Menor importância na documentação

- Modelo pouco conhecido e usado

* Outros modelos (alternativos ou não recomendado)

- Go Horse: é uma metodologia não-metodológica, esse modelo é uma espécie de gambiarra, nesse modelo a execução é priorizada acima de qualquer método, análise e estratégia. Os princípios desse modelo são:

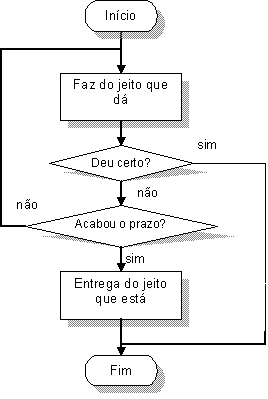
- Se o código está funcionando ele está pronto

- Documentação é perca de tempo

- Se ele funciona, não mexe por que pode quebrar

- O prazo é prioridade

- erros só aparecem quando o cliente tem o produto



Vantagens:

- Entrega no prazo

- Equipe bem relaxada

- Pouca burrocracia

Desvantagens:

- Insatisfação do cliente

- Vários erros técnicos

- Possível retrabalho

- Reputação da equipe

- Infinitas correções