

Engenharia de Software

Profa. Dra. Debora Maria Barroso Paiva



Módulo 1 - Introdução à Engenharia de Software

Unidade 2 - Características dos principais modelos de processos de desenvolvimento de software



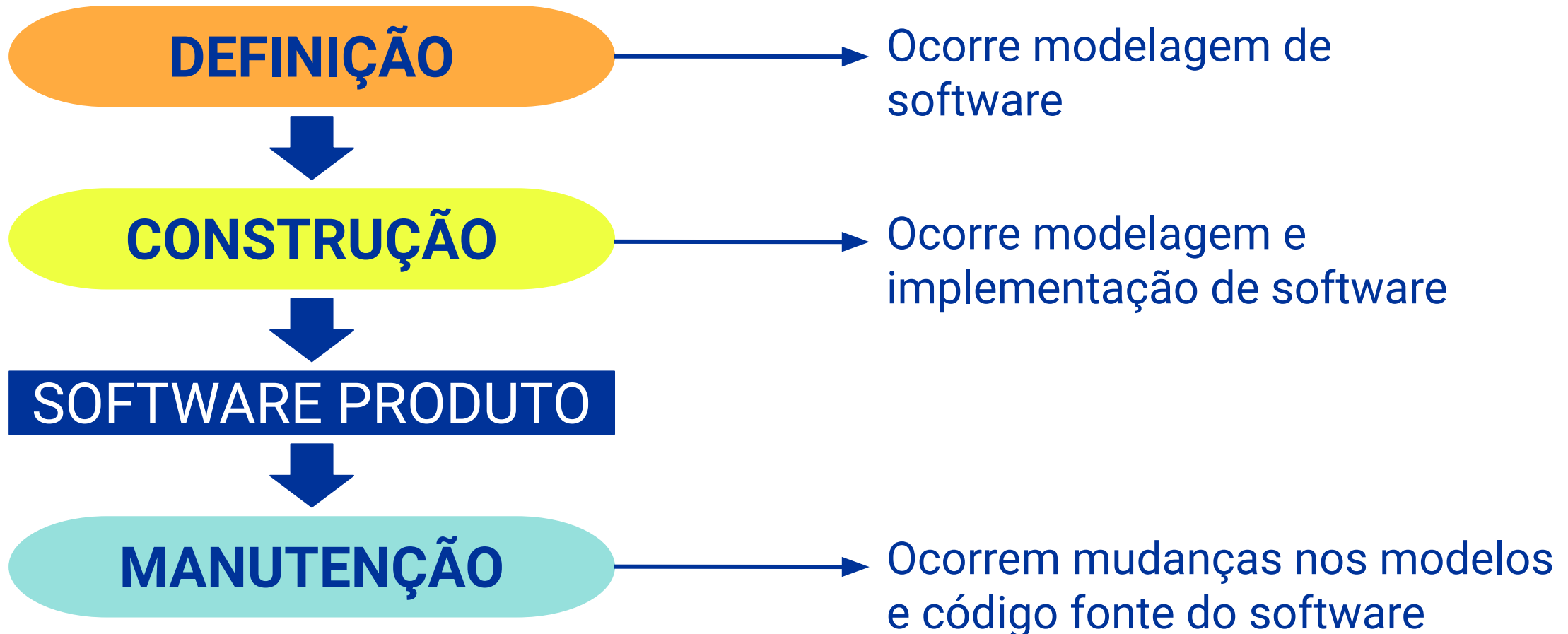
Relembrando...

- **Objetivo:** entender os principais conceitos e definições da engenharia de software e os diferentes modelos que podem ser utilizados para conceber, projetar e desenvolver software.

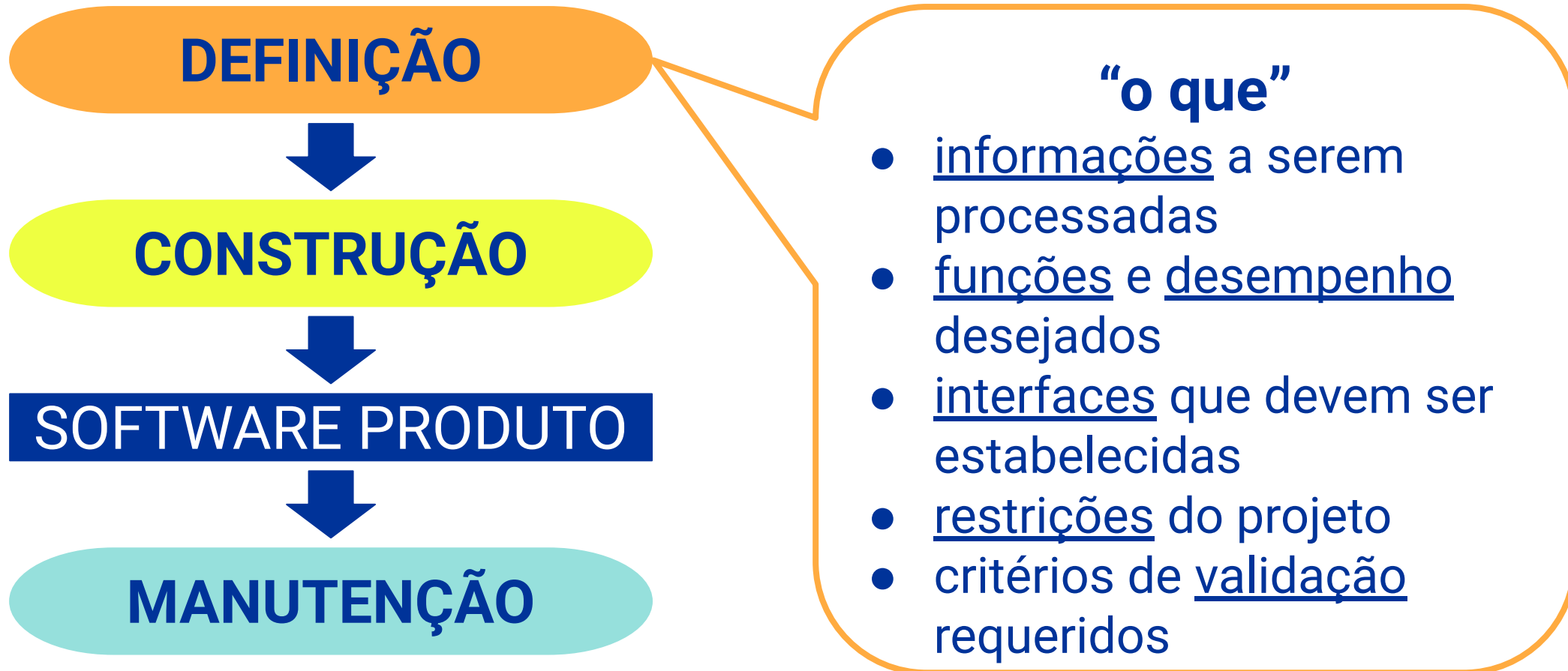
Relembrando...

- **Processos de software:** atividades executadas para criar o produto.

Fases genéricas dos modelos de processo de software



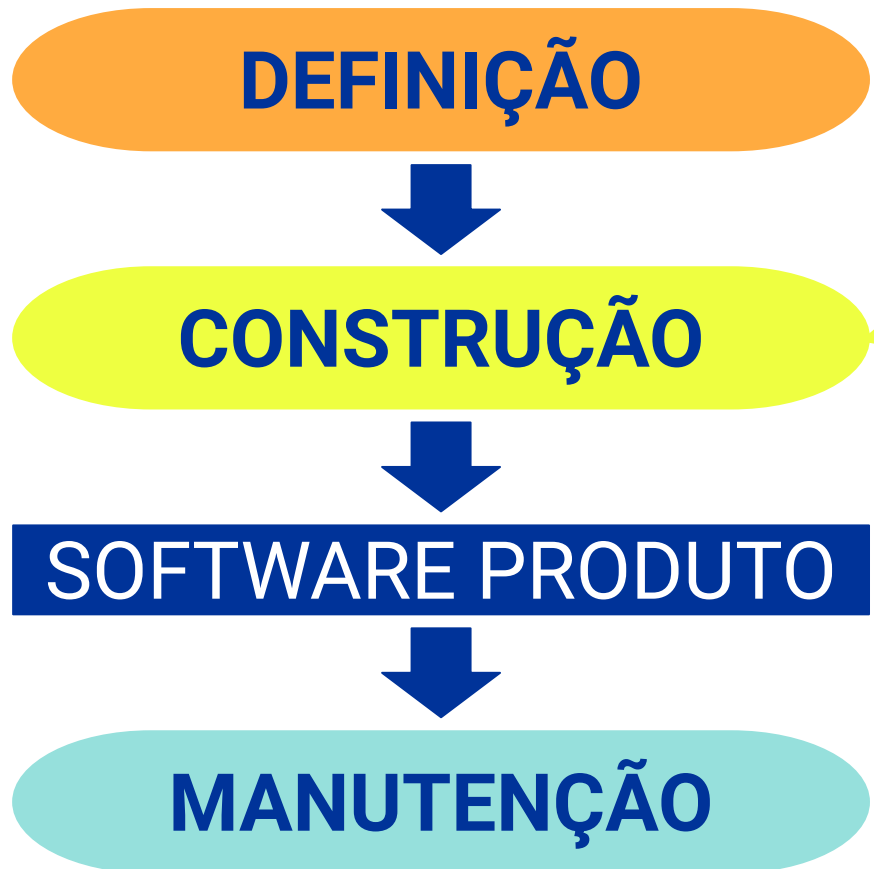
Fases genéricas dos modelos de processo de software



Fases genéricas dos modelos de processo de software



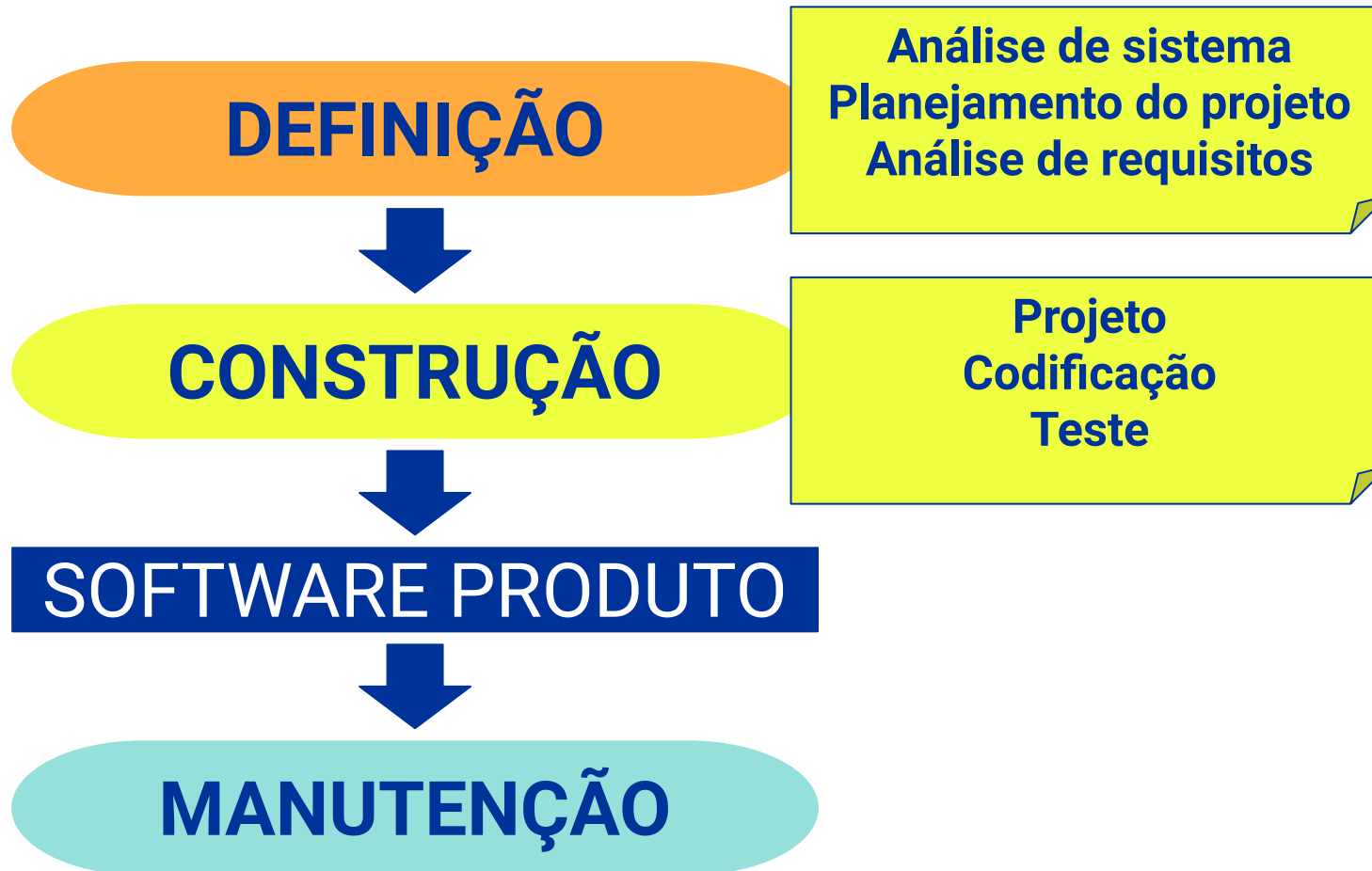
Fases genéricas dos modelos de processo de software



“como”

- como devem ser projetadas a estrutura de dados e a arquitetura do software
- como os procedimentos devem ser implementados
- como o projeto deve ser traduzido para uma linguagem de programação
- como o teste deve ser executado.

Fases genéricas dos modelos de processo de software



Fases genéricas dos modelos de processo de software

DEFINIÇÃO



CONSTRUÇÃO



SOFTWARE PRODUTO



MANUTENÇÃO

“mudanças”

- correções decorrentes de erros - *manutenção corretiva*
- adaptações requeridas decorrentes de alterações no meio ambiente - *manutenção adaptativa*
- melhoramentos relacionados a novas funções desejadas pelo usuário - *manutenção perfectiva*

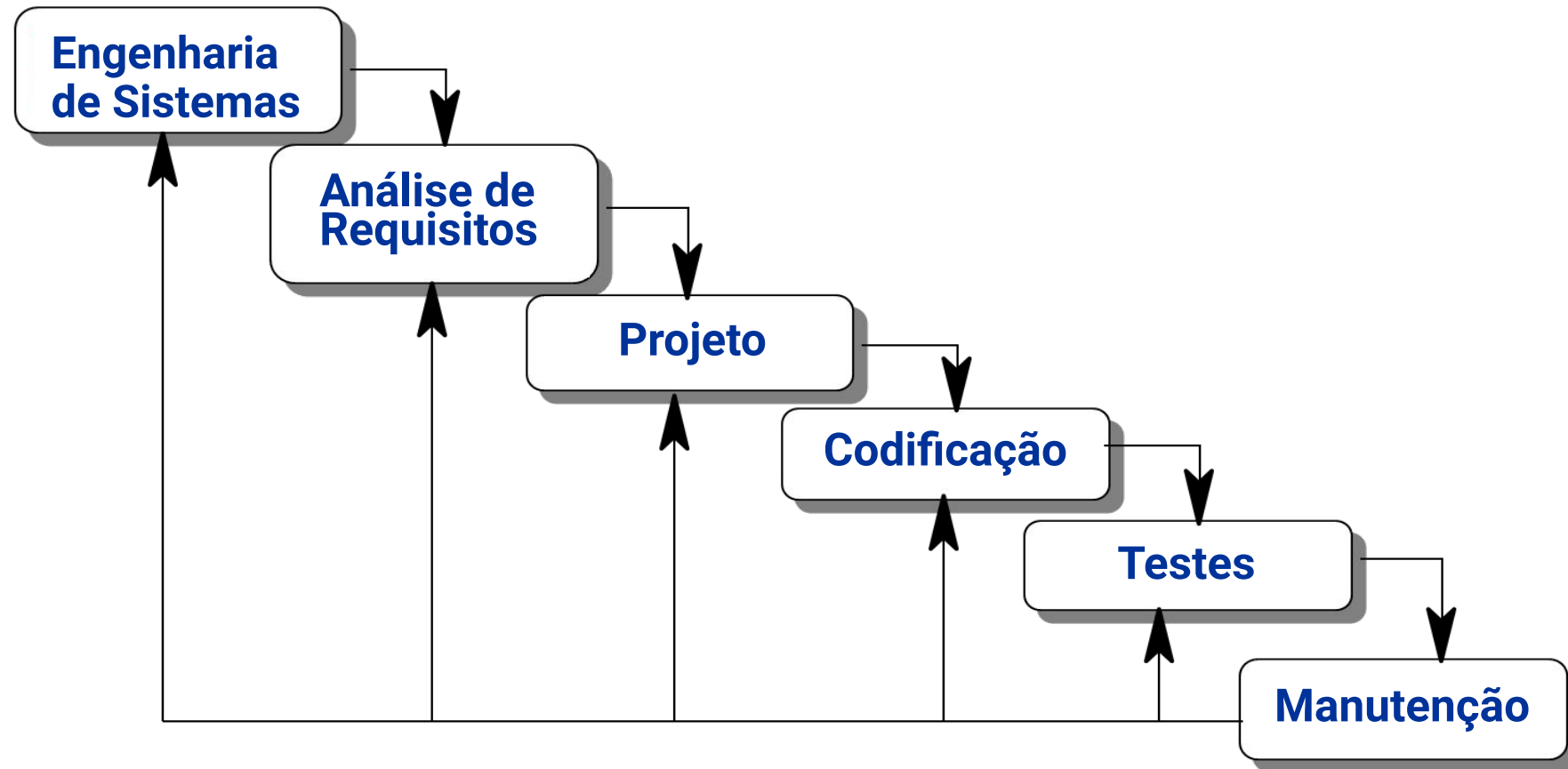
Fases genéricas dos modelos de processo de software



Modelos de processo de software

- Existem vários modelos de processo de software;
- Cada um representa uma tentativa de colocar ordem em um conjunto de atividades complexas;
- Alguns modelos de processo de software que podem ser citados: modelo cascata, de prototipação e espiral.

Modelo cascata



Fonte: Adaptado de Pressman (2021)

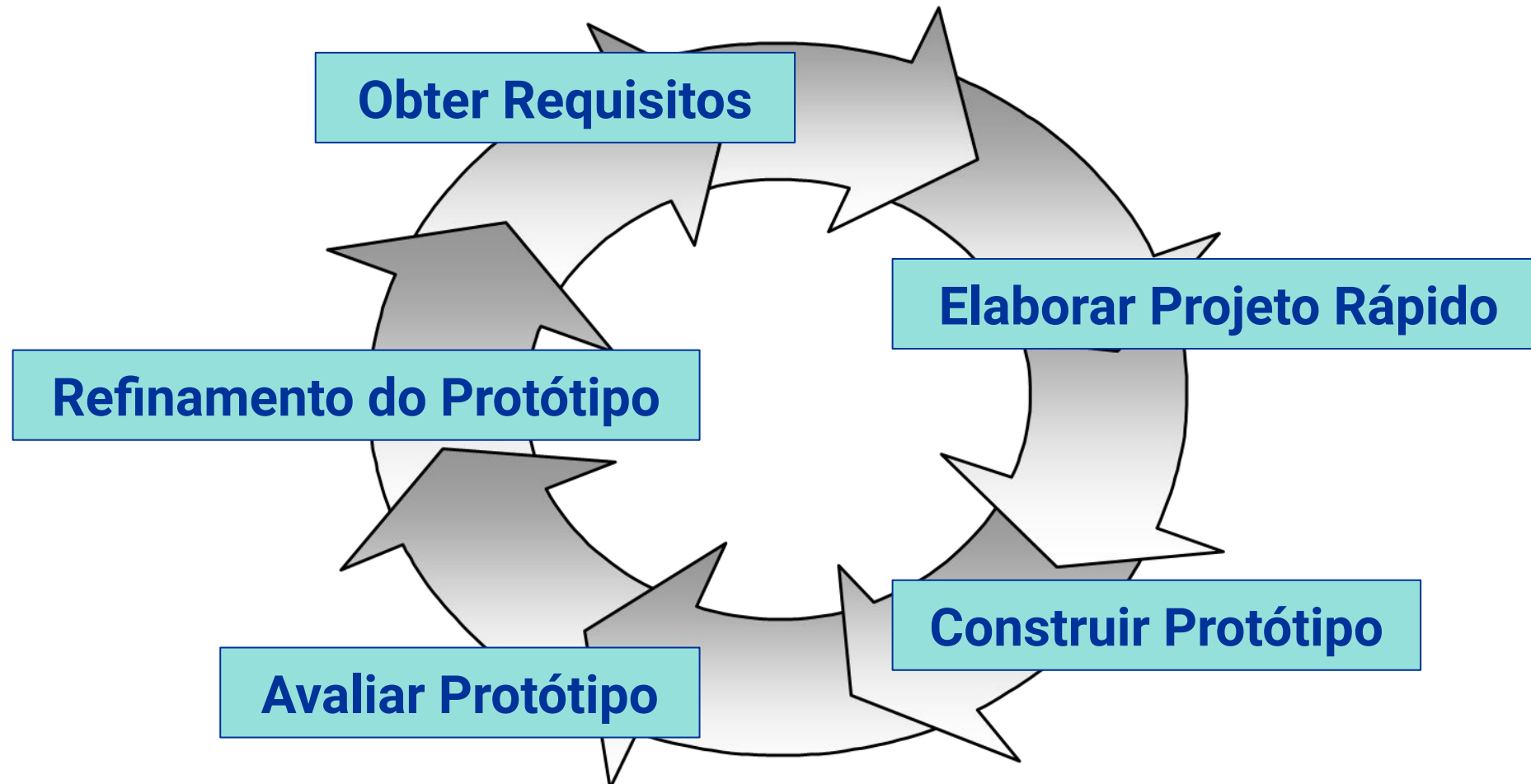
Modelo cascata

- Modelo mais antigo;
- Requer uma abordagem sistemática, sequencial ao desenvolvimento de software;
- O resultado de uma fase se constitui no início da outra.

Modelo de prototipação

- Objetivo: entender os requisitos do usuário;
- Possibilita ao desenvolvedor criar um modelo (protótipo) do software que deve ser construído;
- Apropriado se o cliente define um conjunto de objetivos gerais para o software, mas não identifica detalhadamente esses requisitos.

Modelo de prototipação

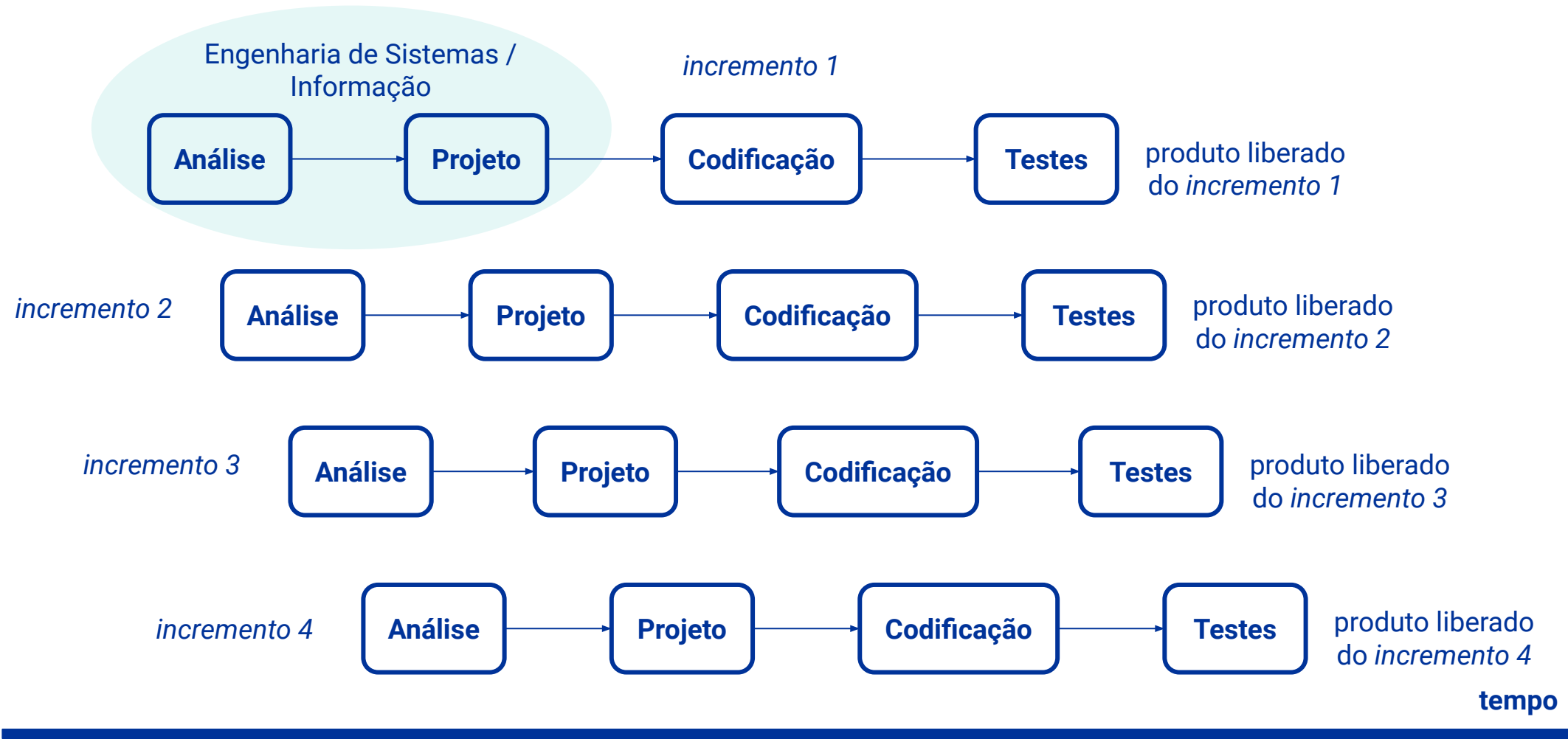


Fonte: Adaptado de Pressman (2021)

Modelo incremental

- Usado quando é difícil estabelecer *a priori* uma especificação detalhada dos requisitos;
- **Objetivo:** trabalhar junto do usuário para descobrir seus requisitos, de forma incremental, até que o produto final seja obtido.

Modelo incremental



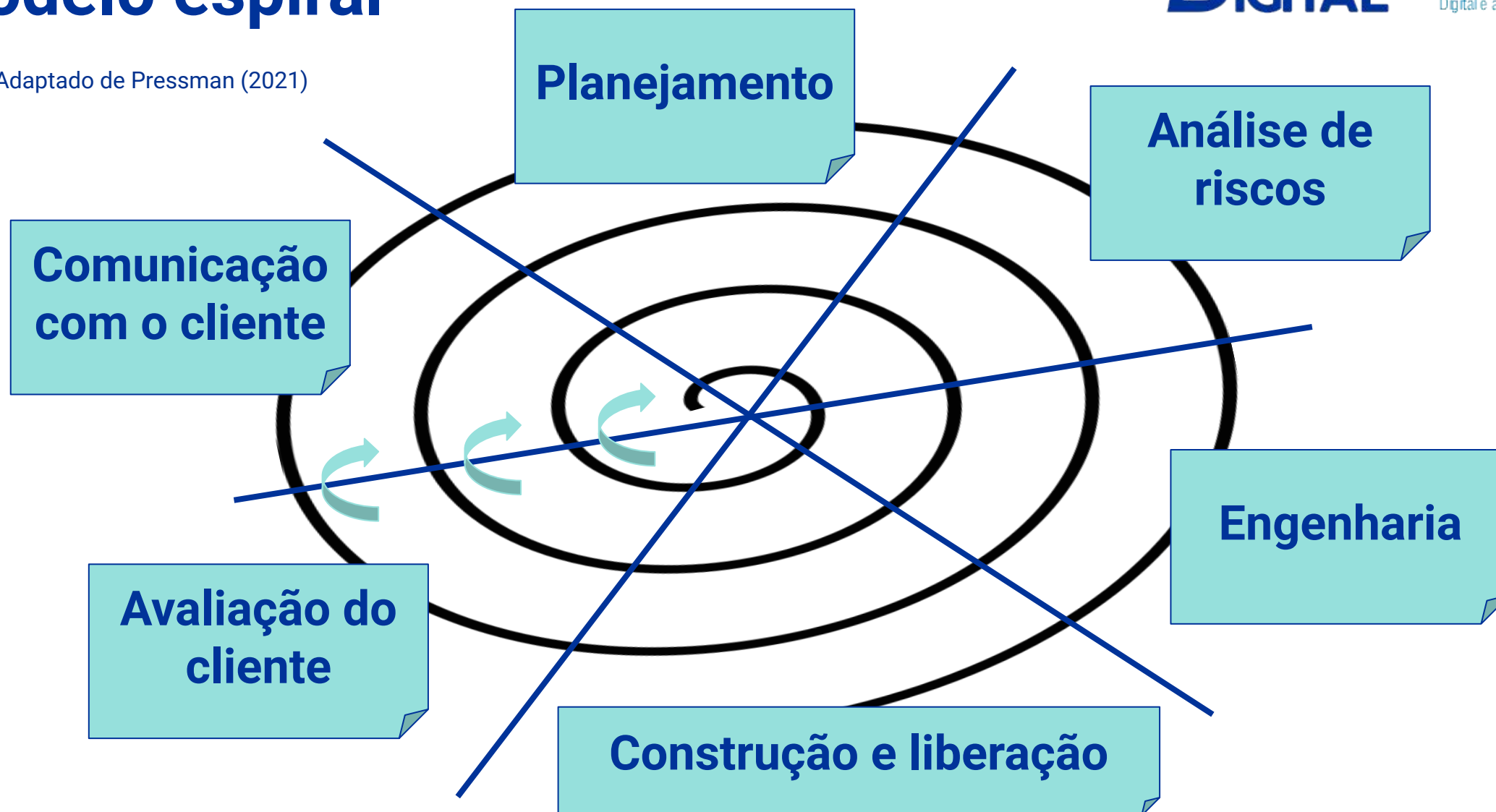
Fonte: Adaptado de Pressman (2021)

Modelo incremental

- Diferentemente do paradigma da prototipação, tem o objetivo de apresentar um produto operacional para cada incremento;
- Os testes podem ser mais efetivos. É mais fácil testar cada versão do sistema do que o sistema todo no final;
- O custo de acomodar mudanças nos requisitos do cliente é reduzido.

Modelo espiral

Fonte: Adaptado de Pressman (2021)



Modelo espiral

- Adiciona uma nova atividade: a ***análise de riscos***;
- Risco é um problema em potencial - pode ocorrer ou não
 - Exemplos:
 - número maior de pessoas acessando o sistema que o planejado;
 - desenvolver um excelente produto que ninguém quer.

Modelo espiral

- Segue a abordagem de passos sistemáticos do modelo cascata, incorporando-os numa estrutura iterativa que reflete mais realisticamente o mundo real;
- Usa uma abordagem que capacita o desenvolvedor e o cliente a entender e reagir aos riscos em cada etapa evolutiva.

Metodologias ágeis

- São uma tentativa de refinar as metodologias iterativas, ao tirar o foco do processo em si e dar mais ênfase para a contribuição das pessoas;
- Criar software útil rapidamente;
- Desenvolvimento incremental;
- Os processos de especificação, projeto e implementação são concorrentes.

Metodologias ágeis

- Não há especificação detalhada de sistema e a documentação do projeto é minimizada ou gerada automaticamente;
- Envolvimento do cliente;
- Pessoas, não processo (os membros da equipe devem desenvolver suas próprias maneiras de trabalhar sem processos prescritivos).

Metodologias ágeis

- Mudanças nos requisitos são bem-vindas, mesmo que tardias;
- Entrega frequente de software funcional;
- Desenvolvimento de projetos em torno de indivíduos motivados;
- Comunicação face a face.

Metodologias ágeis - eXtreme programming

- Voltada para equipes de até 20 pessoas engajadas no desenvolvimento de software cujos requisitos são vagos ou se encontram em constante mudança;
- Constante melhoria do código.

Metodologias ágeis: SCRUM

- Foco: gerenciamento do desenvolvimento iterativo de software;
- Não prescreve o uso de práticas de programação, como a programação por pares;
- Inovação do SCRUM: sua fase central é denominada “ciclos de *sprint*”.

Metodologias ágeis: SCRUM

- Um *sprint* é uma unidade de planejamento na qual o trabalho a ser feito é avaliado, os recursos para o desenvolvimento são selecionados e o software é implementado;
- *Sprints* são de cumprimento fixo, geralmente de duas a quatro semanas;
- O *backlog* do produto é a lista de trabalho a ser feito.

Considerações finais

- Há diversos modelos de processo apresentados na literatura e utilizados na prática pelas empresas de software;
- Os modelos em geral são utilizados conjuntamente, em especial em projetos grandes e complexos;
- Há modelos mais burocráticos (mais documentação) e modelos ágeis (menos documentação).

Referências

PRESSMAN, Roger S; MAXIM, Bruce R. **Engenharia de software**: uma abordagem profissional. 9. ed. Porto Alegre: McGraw Hill, 2021. ISBN 9786558040118. [Disponível na Biblioteca Digital da UFMS.](#)

SOMMERVILLE, Ian. **Engenharia de software**. 10. ed. São Paulo: Editora Pearson, 2018. ISBN 9788543024974. [Disponível na Biblioteca Digital da UFMS.](#)

Licenciamento



Respeitadas as formas de citação formal de autores de acordo com as normas da ABNT NBR 6023 (2018), a não ser que esteja indicado de outra forma, todo material desta apresentação está licenciado sob uma [Licença Creative Commons - Atribuição 4.0 Internacional](https://creativecommons.org/licenses/by/4.0/).