

# Engenharia de Software

## Processo de Software

### O que é a engenharia de software?

É uma das áreas da ciência da computação onde se é sistematizada e dar previsibilidade a atividade de desenvolvimento de software, uma disciplina de engenharia que se preocupa com os aspectos da produção do software, desde sua concepção inicial até sua operação e manutenção.

### O que é software?

Programas de computadores e documentação associada. Os produtos de software podem ser desenvolvidos para um determinado cliente ou para um mercado genérico.

- **software é um produto abstrato e intangível.**

### Quais os atributos de um bom software?

Deve proporcionar a **funcionalidade** e o **desempenho** necessários e deve ser **manutenível**, **usável** e com **dependabilidade (dependability)**.

- **Aceitabilidade:** deve ser aceitável para o tipo de usuário para o qual é projetado, deve ser inteligível, útil e compatível com os outros sistemas utilizados pelo usuário.
- **Dependabilidade e segurança da informação (security):** A dependabilidade inclui uma gama de características, incluindo confiabilidade, segurança da informação (security) e segurança (safety). Não deve causar danos físicos ou econômicos em caso de falha de sistema, assim como ser protegido contra usos e ataques maliciosos.

- **Eficiência:** não deve desperdiçar recursos do sistema, incluindo responsividade, tempo de processamento e utilização de recursos.
- **Manutenibilidade:** deve ser escrito de tal modo que possa evoluir e satisfazer as necessidades mutáveis dos clientes.

### Atividades fundamentais

- **Especificação;**
- **Desenvolvimento;**
- **Validação;**
- **Evolução de software;**

### Desafios enfrentados

Crescente diversidade, com as demandas por menores prazos de entrega e desenvolver software confiável.

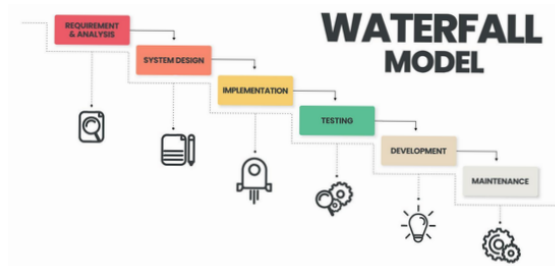
### QUALIDADE -> CUSTO/ORÇAMENTO -> TEMPO/CRONOGRAMA

### Exemplos:

- Aplicações stand-alone;
- Aplicações interativas baseadas em transações;
- Sistemas de controle embarcado;
- Sistemas de processamento em lote (batch);
- Sistemas de entretenimento;
- Sistemas para modelagem e simulação;
- Sistemas de coleta e análise de dados;



## Modelo em cascata



Definição dos requisitos -> Projeto do sistema e do software -  
> Implementação e teste de unidade -> Integração e teste de sistema -> Operação e manutenção.

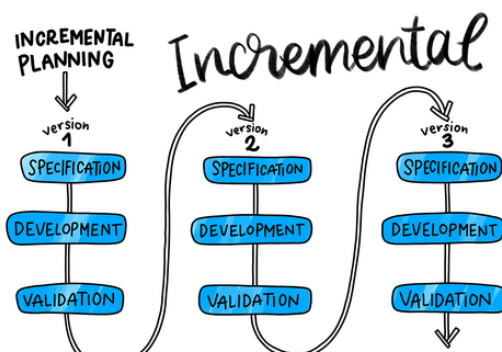
## Modelo de Boehm

### Modelo espiral de Boehm



Divisão de quadrantes temáticos, com múltiplas cascatas, movimento circular (espiral).

## Modelo incremental



Desenvolvimento em iterações incrementais, com revisões e testes em cada iteração, atividades simultâneas.

## Modelo orientado ao reuso

Especificação dos requisitos -> Descoberta e Avaliação do software -> Refinamento dos requisitos -> Adaptações ou Desenvolvimento de componentes -> Integração do sistema.



Videoaula - Modelos de processo de software

## Motivação para desenvolvimento ágil

Ambientes corporativos menores, processos baseados em planos causam muito **overhead**, gasta-se muito com planejamento.

## Dirigido por planos

Engenharia de requisitos -> Especificações de requisitos -> Projeto (design) e implementação.

Definição dos requisitos -> Projeto do sistema e do software -  
> Implementação e teste de unidade -> Integração e teste de sistema -> Operação e manutenção.

## Desenvolvimento ágil

Engenharia de requisitos -> Projeto (design) e implementação.

Definição dos requisitos -> Projeto de implementação teste -> Operação e manutenção.

## Manifesto Ágil, 2001

"Estamos descobrindo maneiras melhores de desenvolver software, fazendo-o nós mesmos e ajudando outros a fazerem o mesmo. Através deste trabalho, passamos a valorizar:

- **Indivíduos e interações** mais que **processos e ferramentas**.
  - **Software em funcionamento** mais que **documentação abrangente**.
  - **Colaboração com cliente** mais que **negociação de contratos**.
  - **Responder às mudanças** mais que **seguir um plano**.
- Ou seja, mesmo havendo valor nos itens à direita, valorizando mais os itens à esquerda."

## Princípios dos métodos ágeis

- **Envolvimento do Cliente;**
- **Acolher as mudanças;**
- **Entrega incremental;**
- **Manter a simplicidade;**
- **Pessoas, não processos;**

## Exemplo:

- Extreme Programming (XP);
- Scrum;
- Test Driven Development (TDD);
- Crystal;
- Adaptative Software Development (ASD);
- Feature Driven Development (FDD);
- Dynamic Systems Development Method (DSDM);
- Agile Unified Process;

### **Dificuldades / Limitações**

- *Cliente deve estar disposto e capaz de passar o tempo com a equipe de desenvolvimento.*
- *Cliente deve ser capaz de representar todas as partes interessadas.*
- *Membros da equipe podem não ter a personalidade adequada.*
- *A organização pode não ter a cultura adequada.*
- *Priorizar mudanças pode ser extremamente difícil.*
- *Manter simplicidade pode ser complicado.*
- *Pode dificultar negociações contratuais.*
- *Depende da maturidade de desenvolvedores.*



**Videoaula** - Métodos de desenvolvimento ágeis

.....