

# Engenharia de Software

Prof. Luiz A. Nascimento

---

---

---

---

---

---

---

## Engenharia de Software

### › Qual a importância da Engenharia de Software?

- O desenvolvimento de um software envolve processos muito complexos. **A engenharia de software estabelece um modelo para se construir um programa.**
- A programação é apenas uma atividade complexa, mas é apenas uma etapa do processo que envolve principalmente o programador. **A engenharia de software envolve diversos intervenientes que vão desde o analista de sistemas até o cliente usuário final.**
- Alguns programas muito simples são feitos por apenas um indivíduo sem precisar utilizar uma técnica estabelecida. **A maioria dos softwares nas empresas são complexos e necessitam de técnicas de desenvolvimento em equipe que envolvem várias pessoas.**

---

---

---

---

---

---

---

## Engenharia de Software

### › Por que para construir um software precisa utilizar técnicas de engenharia?

- O desenvolvimento de um software envolve profissionais especializados em atividades complexas. **Atividades de engenharia como a construção de uma casa necessitam de técnicas de engenharia para a qualidade do produto final.**
- É muito difícil determinar o prazo final de entrega de um software. **Atividades de engenharia também necessitam de engenheiros para elaborar um cronograma para definir a entrega do produto.**
- Os custos de um software é muito difícil de determinar. **É necessário fazer um estudo com um orçamentista para definir os cálculos das despesas de uma obra.**

---

---

---

---

---

---

---

## Engenharia de Software

### › O que é um software?

- Um produto que os profissionais de software constroem e mantêm ao longo do tempo.

Pressman

- Conjunto de programas (arquivos de configuração, documentação, etc.) .

Summerville

---

---

---

---

---

---

---

## Engenharia de Software

### › Existem tipos diferentes de software?

- Um software pode ser classificado como:

- Básico
- Aplicativo
- Sistemas
- Científico e de Engenharia
- Embutido
- Computador Pessoal
- Aplicações Web
- Inteligência Artificial
- Comercial

---

---

---

---

---

---

---

## Engenharia de Software

### › Por que dizem que estamos vivendo uma crise de software?

- Porque existem muitos problemas no desenvolvimento do software.
- A crise do software está relacionada à problemas que são encontrados no desenvolvimento de software de computador que não se limitam ao não funcionamento de forma adequada. Envolve também outros problemas:
  - Como se desenvolve um software;
  - Como mantemos um volume crescente de software existente;
  - Como administrar a crescente demanda por mais software.

---

---

---

---

---

---

---

## Engenharia de Software

- › O Software atualmente é mais importante e faz mais diferença na qualidade de um sistema do que o Hardware (Era da Informação).
  - *completeza* da informação
  - *usabilidade*
  - *otimização*
  - Interface para web
  - inteligência
  - funcionalidade
  - compatibilidade
  - suporte

---

---

---

---

---

---

---

## Engenharia de Software

- › O ideal é que se crie um software economicamente viável, que seja confiável e que funcione eficientemente em máquinas reais.
- › Deve-se ser possível construir um software com várias versões por várias pessoas.”

---

---

---

---

---

---

---

## Engenharia de Software

- › **Equipe básica de desenvolvimento:**
  - › O **Engenheiro de software** é responsável por todas as diretrizes do projeto e desenvolvimento de sistemas de software.
  - › O **Analista de sistemas** é responsável por extrair conhecimento dos usuários e clientes para desenvolvimento do software. Especifica o software através de um modelo que o programador entenda.
  - › **Programador** é o responsável em codificar o software de acordo com o projeto em uma linguagem de programação.

---

---

---

---

---

---

---

## Engenharia de Software

### *Ensino no desenvolvimento de Software*

#### ▶ Escolas ensinam a criar pequenos programas

- Não há tempo para ensinar (horas aula)
- Abordam tarefas fáceis de executar
- Mais fácil de aprender

#### ▶ É difícil ensinar softwares complexos

- Não há tempo disponível
- Dificuldade em aprender
- Necessita trabalho supervisionado em grupo
- Pessoas aprendem na prática nas empresas (+ - 1 ano)

---

---

---

---

---

---

---

## Engenharia de Software

### *Definições de Engenharia de software*

#### ▶ Disciplina que lida com a construção de sistemas de software:

- Flexíveis
- Modulares
- Robustos
- Confiáveis
- Usáveis
- Adequados ao contexto onde são inseridos.

---

---

---

---

---

---

---

### *Introdução à Engenharia de Software*

#### ▶ Pequenos Programas x Programas Complexos

#### ▶ Pequenos programas são relativamente fáceis de construir.

- Apenas uma pessoa desenvolve
- Poucas linhas de código
- Pouca manutenção

#### ▶ Programas complexos são difíceis de construir.

- Necessidade de decomposição em módulos
- Interface entre módulos
- Grande quantidade de códigos
- Várias pessoas desenvolvem

---

---

---

---

---

---

---

## Engenharia de Software

Programação	Engenharia
Projeto simples e pequeno	Projeto grande e complexo
Uma pessoa	Equipes
O que a pessoa quer	O que os clientes querem
Um produto	Uma família de produtos
Poucas modificações	Modificações em paralelo
Tempo de vida curto	Tempo de vida longo
Barato	Caro

---

---

---

---

---

---

---

---

## Engenharia de Software

### *Características do Software*

- ▶ Atualmente os softwares mais complexos são desenvolvidos com técnicas de engenharia.
- ▶ O desenvolvimento necessita de projeto e de gerenciamento do processo.
- ▶ Falhas em software indicam erro de projeto ou de implementação.
- ▶ A manutenção do software é mais complicada que a do hardware.
- ▶ Software não desgasta (Normalmente deve ser projetado para o maior tempo de vida útil possível).

---

---

---

---

---

---

---

---

## Engenharia de Software

### *Características do Software*

- ▶ A maioria dos softwares são construídos sob medida e não a partir de componentes existentes.
  - Com a tecnologia de Orientação a Objetos e a reusabilidade de códigos está mudando este cenário.

---

---

---

---

---

---

---

---

## Engenharia de Software

### *Motivos para a Engenharia de Software*

- › Os prazos são excedidos.
- › Os custos são cada vez maiores por causa de sua complexidade.
- › Muitos erros de software são descobertos pelo cliente depois de sua entrega.
- › Existe grande dificuldade em medir o progresso do desenvolvimento de software.

---

---

---

---

---

---

---

## Engenharia de Software

### *Algumas causas dos problemas*

- › Não é dedicado tempo suficiente para coletar dados sobre o desenvolvimento do software.
- › Muitas coisas são estimadas.
- › A comunicação entre o cliente e o desenvolvedor é falha.
- › Falta de testes
- › Gerentes com pouco know-how
- › Pouco treinamento para o pessoal
- › Falta de investimento em Engenharia de Software

---

---

---

---

---

---

---

## Engenharia de Software

### *Mitos dos softwares - Visão cliente*

- › Colocar mais programadores agiliza o processo e evita atrasos.
- › Utilizar um computador mais potente resolve os problemas de desenvolvimento.
- › Uma declaração geral e óbvia é suficiente para iniciar o desenvolvimento.
- › Mudanças podem ser feitas sem problemas na fase de desenvolvimento.
- › Os programadores conseguem fazer quase tudo.

---

---

---

---

---

---

---

## Engenharia de Software

### *Mitos dos softwares - Visão desenvolvedor*

- › Um programa está terminado ao funcionar.
- › Quanto mais cedo escrever um código mais rápido o programa será terminado.
- › Só é possível avaliar a qualidade de um software em funcionamento.
- › A única coisa a ser entregue em um sistema é o programa.

---

---

---

---

---

---

---

## Engenharia de Software

- › Uso de técnicas de engenharia para o desenvolvimento de um software com qualidade.
- › Existe a necessidade de se adotar uma metodologia para o desenvolvimento e o gerenciamento dos processos.

### Sistema sem projeto definido por humanos

- Ex. Casas em uma favela ou plantas na floresta



### Sistema com projeto definido por humanos

- Ex. Conjunto residencial ou plantas na fazenda.



---

---

---

---

---

---

---

## Engenharia de Software

### Usuários

- Pouco conhecimento de suas necessidades
- Não conhecem/dominam as capacidades e limitações dos computadores
- Omissão de informações "óbvias"
- Usuários com opiniões diferentes
- Linguagem diferente do desenvolvedor

### Desenvolvedor

- Pouco conhecimento sobre o domínio do problema
- Orçamento
- Linguagem diferente do usuário
- Limitações técnicas

---

---

---

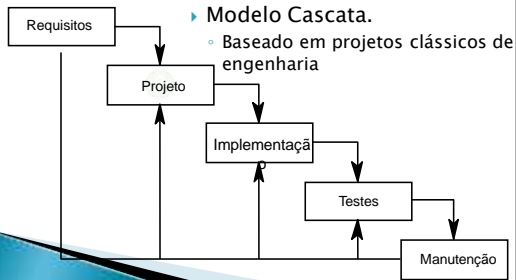
---

---

---

---

## Modelo clássico de desenvolvimento de Software – Modelo Cascata



---

---

---

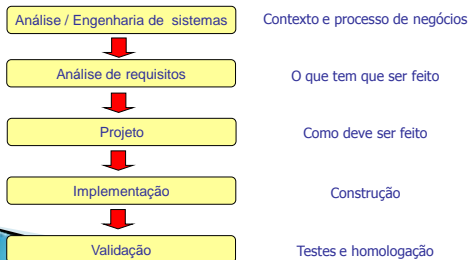
---

---

---

---

## Modelo clássico de desenvolvimento de Software – Modelo Cascata



---

---

---

---

---

---

---

## Modelo clássico de desenvolvimento de Software – Modelo Cascata

### Modelo Cascata – Vantagens

- Desacoplamento da equipe
- Fases do produto bem definida
- Melhor visão geral do processo

### Modelo Cascata – Desvantagens

- Não trabalha em paralelo
- Etapas complexas
- Dificuldade de retrabalho
- Pouca interação com o usuário
- Dificuldade da declaração de todas as exigências pelo cliente
- Pouca paciência do cliente

---

---

---

---

---

---

---