

# Engenharia de Software

**Aula: Dificuldades no Desenvolvimento Software**

# Desenvolvimento de Software



- Como são desenvolvidos os softwares?
- Estamos sendo bem sucedidos nos softwares que construímos?
- Quão difícil é construir software?

# Desenvolvimento de Software



# O Produto de Software

## ■ Engloba:

- ✓ **PROGRAMAS** que quando executados produzem a função e o desempenho desejados.
- ✓ **ESTRUTURAS DE DADOS** que possibilitam que os programas manipulem adequadamente a informação.
- ✓ **DOCUMENTOS** que descrevem a operação e o uso dos programas.

# O Desenvolvimento Software

## Consiste:

- ✓ Compreender bem o problema do cliente.
- ✓ Desenvolver uma solução.
- ✓ Garantir que ela funcione.
- ✓ Transformar a solução em um software.
- ✓ Entregar o software ao cliente no custo e prazo combinados.
- ✓ Oferecer treinamento aos usuários.
- ✓ Realizar alterações quando o cliente pedir.
- ✓ Manter o controle da versão do software que cada cliente possui.

# O Desenvolvimento Software Consiste:

- **Tudo isso enquanto**

- ☐ Negocia preços e prazos;
- ☐ Paga o salário dos funcionários;
- ☐ Capacita os funcionários;
- ☐ Se mantém atualizado com o mercado tecnológico.

- **E ainda** busca maneiras de fazer tudo isso cada vez **melhor**, mais **rápido** e **barato** que a concorrência.

# Dificuldades no Desenvolvimento Software

Projetar um sistema que atenda todas as necessidades e expectativas dos clientes, no prazo estipulado e no custo previsto é uma tarefa extremamente complexa.

- Muitos dos softwares entregues ao cliente **falham em não atender as necessidades e expectativas** dos usuários.
- Além disso, vários softwares são **entregues fora do prazo** combinado e com os **custos mais altos** que os estimados inicialmente.

# Dificuldades no Desenvolvimento Software

- **25% dos projetos são cancelados;**
  - ❑ O tempo de desenvolvimento é bem maior do que o estimado;
- **75% dos sistemas não funcionam como planejado;**
  - ❑ A manutenção e reutilização são difíceis e custosas;
- ❑ **Crise do SOFTWARE**



# CRISE DO SOFTWARE

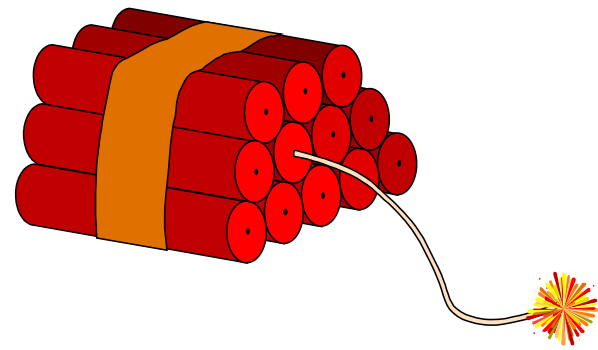
## • No princípio era assim...

- O desenvolvimento de software antes da década de 70 era feito sem o uso de metodologias;
- SW era mais barato do que o HW;
- Má formação de pessoal de desenvolvimento, baseado apenas na necessidade;
- Concepção “privada” de produção;
- Criação de **mitos** e maus hábitos.

# Crise do Software

# Crise do Software

- Refere-se a um conjunto de problemas encontrados no desenvolvimento de software
  - ✓ estimativas de prazo e de custo ↑
  - ✓ produtividade das pessoas ↓
  - ✓ qualidade de software ↓
  - ✓ software difícil de manter ↑



# Crise do Software

- As **estimativas** de **prazo** e de **custo** frequentemente são imprecisas.
- Frequente **insatisfação do cliente** com o sistema concluído.
- A **qualidade do software** é, em geral, abaixo da adequada.
- O software existente é muito **difícil de manter**.
- Falhas das **pessoas** responsáveis pelo desenvolvimento de software.

# **ALGUNS MITOS DE DESENVOLVIMENTO**

## MITO no. 1

*Uma simples declaração de objetivos é suficiente para se começar a escrever programas! – Podemos preencher os detalhes mais tarde*

- A falta ou uma fraca definição é a principal causa de esforços para CONSERTAR software;
- Retrabalho considerável;
- Aumento do custo.



não era isto que  
esperava



Como o cliente explicou



Como o lider de projeto entendeu



Como o analista planejou



Como o programador codificou



O que os beta testers receberam



Como o consultor de negocios descreveu



Valor que o cliente pagou



Como o projeto foi documentado



O que a assistencia tecnica instalou



Como foi suportado



Quando foi entregue

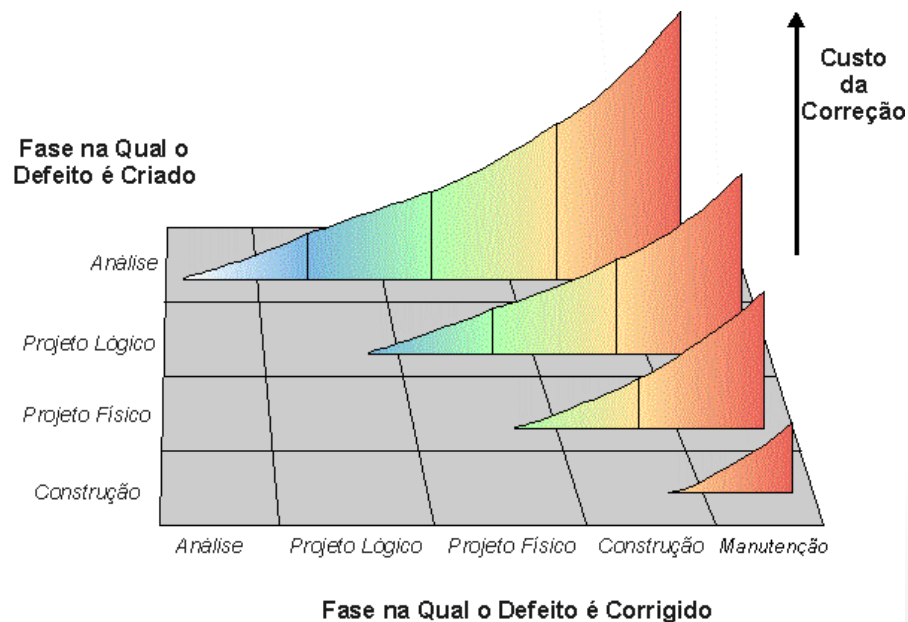


O que o cliente realmente necessitava

## MITO no. 2

***Qualquer mudança nos requisitos é fácil de ser acomodada, pois o software é flexível!***

- É flexível em relação ao hardware, mas os efeitos das mudanças variam no instante no qual a mudança é introduzida.





# Resposta à Crise de Software

## Engenharia de Software

A aplicação de uma abordagem **sistemática**, **disciplinada** e possível de ser **medida** para o **desenvolvimento**, **operação** e **manutenção** do software (*IEEE*).



**Processo de  
Software**