



Fundamentos de Desenvolvimento de Softwares



Conversa Inicial



2
34



A engenharia de software e o engenheiro de software

- Nesta aula, veremos o que é a Engenharia de Software e quais são as funções de um engenheiro de software. Abordaremos também questões relativas ao surgimento da Engenharia de Software, bem como os mitos e a importância dessa área

■ Temas abordados

- **Crise do *software***
- **Engenharia de *software***
- **Engenheiro de *software***
- **Mitos do desenvolvimento de *software***
- **Importância da engenharia de *software***

Crise do *software*



5
34



Crise do *software*

- Refere-se ao problema do *software* não acompanhar o desenvolvimento do *hardware*
- Conceito de Engenharia de *Software*: inicialmente proposto em 1965, durante conferência sobre a crise do *software*



Bugs mais famosos

Spreadthesign/
Shutterstock



▪ *Bug*

- Erro na interpretação de um requisito
- Erro de sintaxe no código
- A causa (ainda desconhecida) da queda de um sistema

Naddy/
Shutterstock



- 1962: falha na sonda

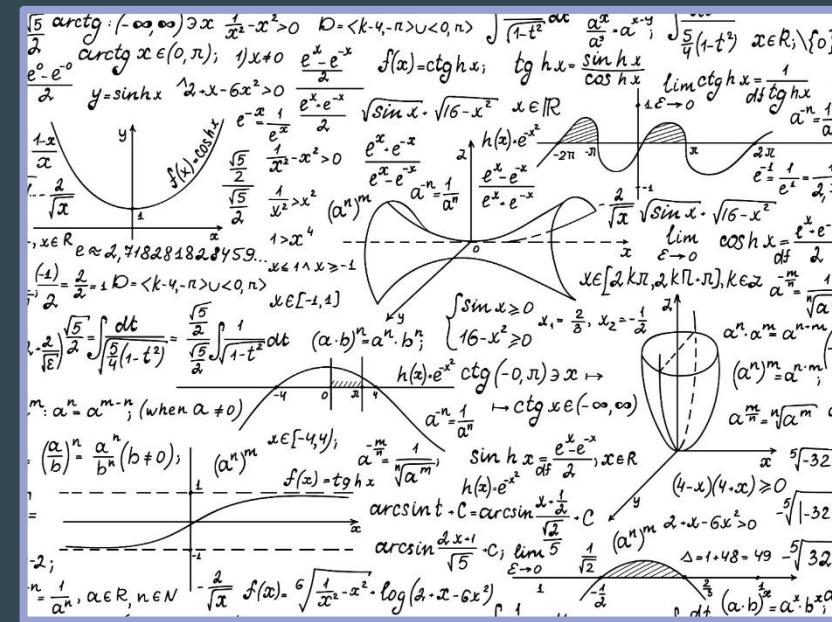
Mariner 1

- Sonda para o planeta
Vênus

- Desvio do curso
preestabelecido

- Destruída quando se
encontrava sobrevoando
o Atlântico

- O problema: fórmula
escrita a lápis que depois
foi digitada errada no
computador de cálculo



Marina Sun/Shutterstock

- 1985-1987: acelerador médico Therac-25
 - Acelerador linear empregado nos hospitais na década de 80 para tratar tumores
 - O problema: devido a uma falha de programação, a máquina emitia 100 vezes mais energia do que a requerida
 - Consequência: cinco pacientes morreram e várias pessoas sofreram os efeitos de ficarem expostas a uma elevada radiação

- 1993: divisão de números com ponto flutuante no Pentium
 - Exemplo:
 - ✓ $4195835,0 / 3145727,0 = 1,33374$
 - ✓ $4195835,0 / 3145727,0 = 1,33382$
 - A Intel foi obrigada a trocar entre três e cinco milhões de chips
 - Prejuízo: mais de meio bilhão de dólares

Erro de
0.006%

Engenharia de *software*



12
34



- Engenharia de *software*
 - 1. Aplicação de uma abordagem sistemática, disciplinada e quantificável no desenvolvimento, na operação e na manutenção de *software*, isto é, a aplicação de engenharia ao *software***
 - 2. O estudo de abordagens como definido em 1**

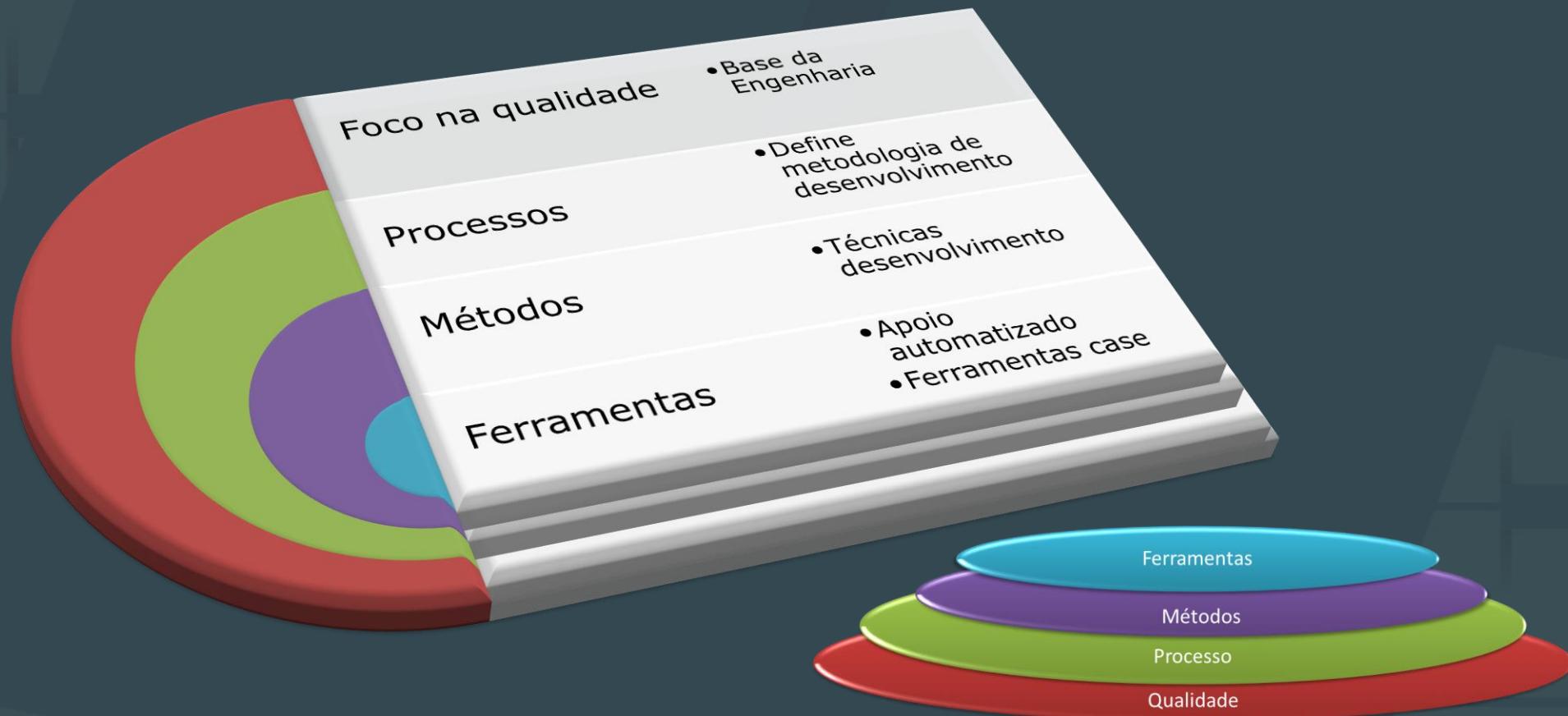
- Engenharia de *software* é uma disciplina de engenharia relacionada a todos os aspectos de produção de *software*

■ A engenharia de *software*

- Abrange um processo
- Um conjunto de métodos
- Ferramentas

Software de
altíssima
qualidade

Camadas da engenharia de *software*



Engenheiro de *software*

- Os engenheiros de *software* aplicam o processo de Engenharia de Software
- Recentemente, em 2018, a Engenharia de *Software* foi regulamentada no Brasil, por meio da Resolução n. 1.100, regulamentada pelo Conselho Federal de Engenharia e Agronomia (Confea) e vinculada aos Conselhos Regionais de Engenharia e Agronomia (CREAs)



Murilo Mazzo/Shutterstock

O que não é

- O engenheiro de *software* não é um desenvolvedor que trabalha nas atividades de análise e produção de código

Engenheiro de *software*



Engenheiro de *software*

- Escolhe, especifica os processos de planejamento, gerência e produção a serem implementados
- Acompanha e avalia o desenvolvimento
- Verifica se o processo é executado de forma eficiente e efetiva
- Reavalia o processo continuamente



Analista

- Responsável pela compreensão do problema
- Levantamento dos requisitos e da modelagem
- Descobre do que o cliente precisa



Programador

- Construção da solução física a partir das especificações
- Conhecimento profundo da linguagem e do ambiente de programação e bibliotecas
- Conhecimento sobre teste e depuração

Engenheiro de software

■ O engenheiro de *software* não deve se preocupar apenas com questões técnicas

Confidencialidade

- Respeitar confidencialidade de funcionários e clientes

Competência

- Não aceitar trabalhos fora de sua competência

Direitos sobre propriedade intelectual

- Estar ciente das leis locais
- Assegurar proteção da propriedade intelectual de funcionários e clientes

Mau uso dos computadores

- Não utilizar habilidades técnicas para fazer mau uso dos computadores

Mitos do desenvolvimento de *software*



Mitos do desenvolvimento de *software*

- Crenças infundadas sobre o *software* e sobre o processo utilizado para criá-lo
- Mitos de gerenciamento
- Mitos do cliente
- Mitos dos profissionais da área

Mitos do gerenciamento

- Já temos um livro cheio de padrões e procedimentos para desenvolver *software*
- Se o cronograma atrasar, acrescentamos mais programadores
- Se eu decidir terceirizar o projeto de *software*, posso simplesmente relaxar e deixar a outra empresa realizá-lo

Mitos do cliente

- Uma definição geral dos objetivos é suficiente para começar a escrever os programas, os detalhes podem ser preenchidos posteriormente
- Os requisitos de software mudam. As mudanças podem ser facilmente assimiladas, pois o software é flexível

Mitos dos profissionais da área

- **Mito**
 - **Uma vez que o programa foi feito e colocado em uso, nosso trabalho está terminado**
- **Realidade**
 - **Entre 60 e 80% de todo o esforço será despendido após a entrega do software ao cliente pela primeira vez**

- **Mito**
 - Até que o programa esteja “em execução”, não há como avaliar sua qualidade
- **Realidade**
 - Existem formas de avaliar a qualidade antes da implementação do sistema

- **Mito**
 - O único produto passível de entrega é o programa em funcionamento
- **Realidade**
 - Vários outros artefatos são necessários: modelos, documentos, planos etc.

■ Mito

- A engenharia de *software* nos fará criar documentação volumosa e desnecessária e, invariavelmente, vai nos retardar

■ Realidade

- O objetivo é criar um produto de qualidade. Uma qualidade melhor leva à redução do retrabalho

Importância da engenharia de *software*



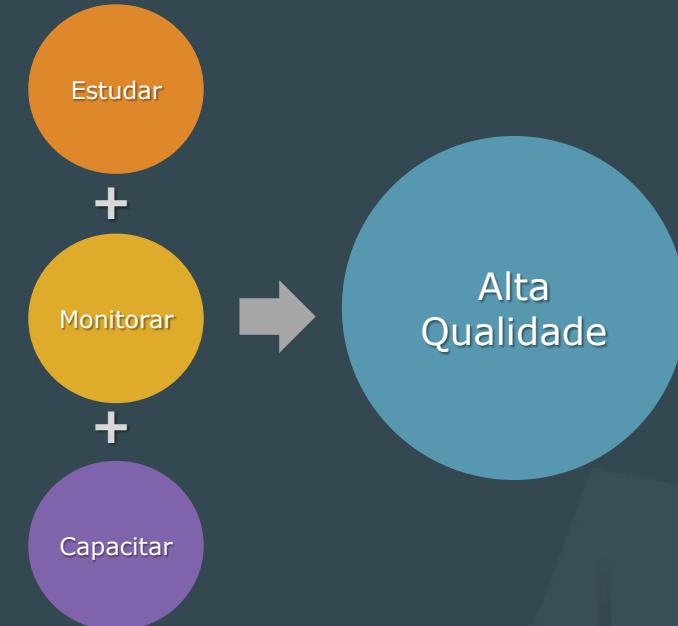
30
34



Importância da engenharia de *software*

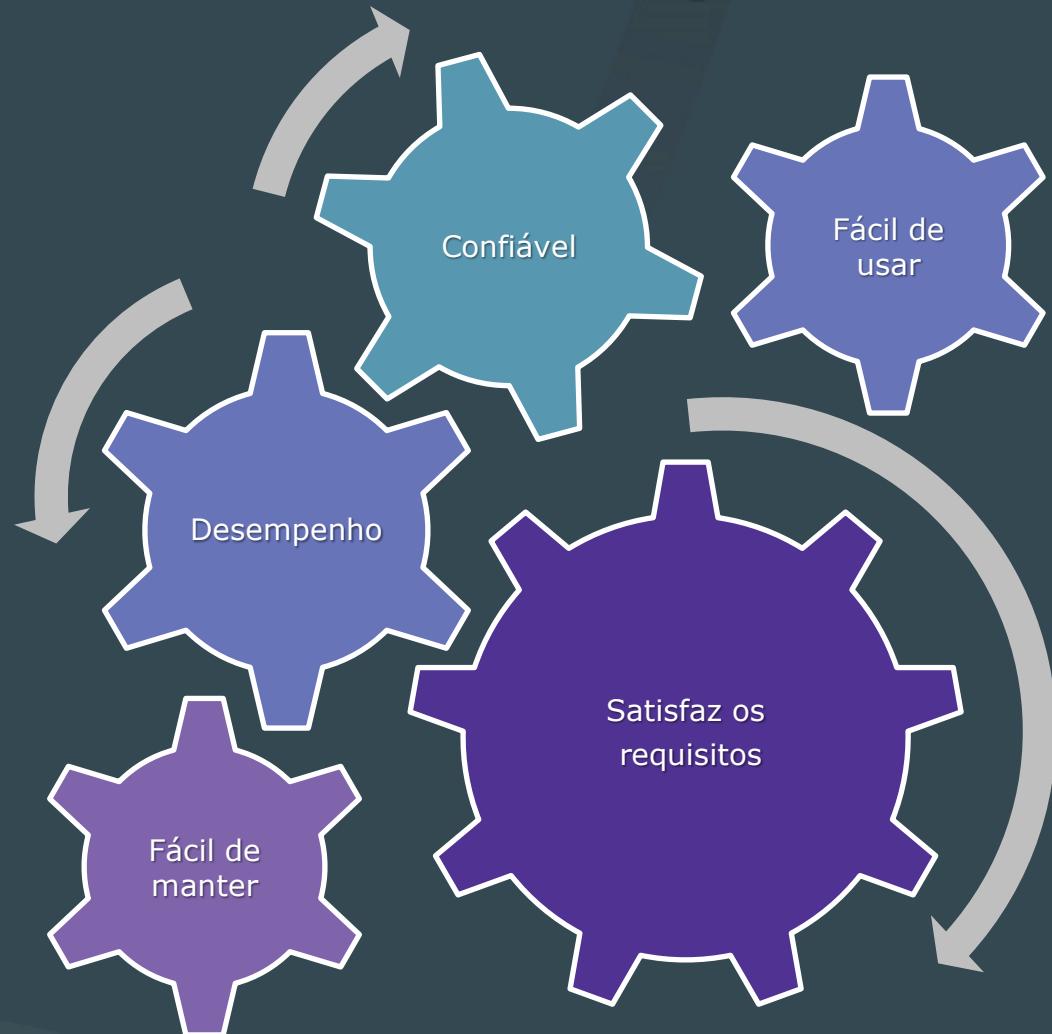
- **Engenharia de *software***

- **Estudar**
 - **Monitorar**
- } Atividades do processo de desenvolvimento



- **Capacita para o desenvolvimento de sistemas complexos dentro do prazo**

Software de qualidade



Importância da Engenharia de *Software*

- Solução
- Pessoas
- Cronogramas

Planejamento

Qualidade

- Produto
- Atendimento das necessidades

- Expectativas do cliente

Requisitos

Desenvolvimento

- Escolha do processo ideal

- Sistema planejado
- Código limpo
- Menos bugs

Manutenção

Reflexão

- “Toda vez que pensar ‘não temos tempo para engenharia de *software*’, pergunte a si mesmo: ‘teremos tempo para fazer de novo?’” (Pressman, 2016, p. 25)

X

Fechar