



# **ENGENHARIA DE SOFTWARE 2**

**Prof. Raul Sidnei Wazlawick**

**UFSC-CTC-INE**

# VISÃO GERAL

## *Processo de desenvolvimento de software*

1. Introdução
2. Processo
3. Modelos de processo prescritivo
4. Modelos ágeis
5. Processo unificado

## *Parte II: Planejamento e gerência de projetos*

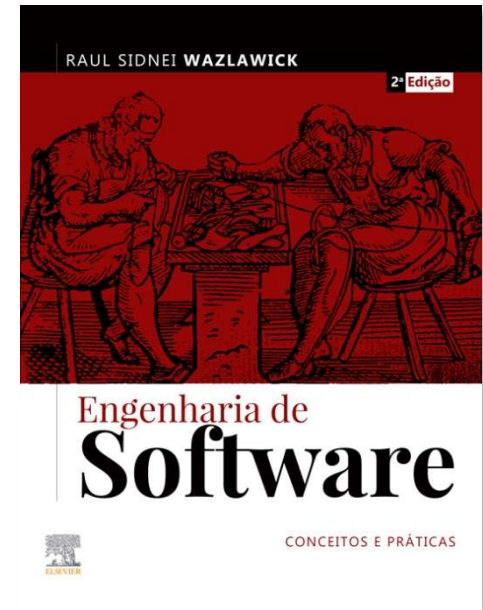
6. Planejamento
7. Estimação de esforço
8. Riscos
9. Gerenciamento de projeto de software
10. Gerenciamento de configuração e mudança

## *Parte III: Qualidade*

11. Qualidade de produto de software
12. Qualidade de processos
13. Teste
14. Manutenção e evolução de software

## *Parte IV: Especificação formal*

1. Modelagem Conceitual
2. Modelagem Funcional



# INTRODUÇÃO



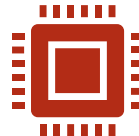
# CONTEÚDO



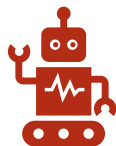
A Crise dos  
Desenvolvedores de  
Software Menos  
Preparados



Os Eternos Mitos



(In)Definição de  
Engenharia de Software



O Engenheiro de  
Software



Evolução da Engenharia  
de Software



# A CRISE DOS DESENVOLVEDORES DE SOFTWARE MENOS PREPARADOS



- O termo “crise do software” foi usado pela primeira vez com impacto por Dijkstra (1972).
- Quais os problemas relatados por ele?



# PROBLEMAS RELATADOS POR DIJKSTRA EM 1972

Embora a Engenharia de Software tenha evoluído como ciência, sua aplicação na prática ainda não é universal.

---

Projetos que estouram o cronograma.

---

Projetos que estouram o orçamento.

---

Produto final de baixa qualidade ou não atendendo aos requisitos.

---

Produtos não gerenciáveis e difíceis de manter e evoluir.





# AINDA EXISTE A CRISE DO SOFTWARE?



# QUAIS AS FRUSTRAÇÕES DOS USUÁRIOS?





# FRUSTRAÇÕES DOS USUÁRIOS



ERROS E FALHAS  
INADMISSÍVEIS EM  
SISTEMAS  
ENTREGUES



INSEGURANÇA EM  
USAR TAIS  
SISTEMAS



CONSTANTE  
NECESSIDADE DE  
MANUTENÇÃO



ALTO CUSTO



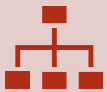
# QUAIS AS FRUSTRAÇÕES DOS DESENVOLVEDORES?



# FRUSTRAÇÕES DOS DESENVOLVEDORES



Sentem que sua produtividade é baixa em relação ao seu potencial



Lamentam a falta de qualidade no produto gerado por seu trabalho



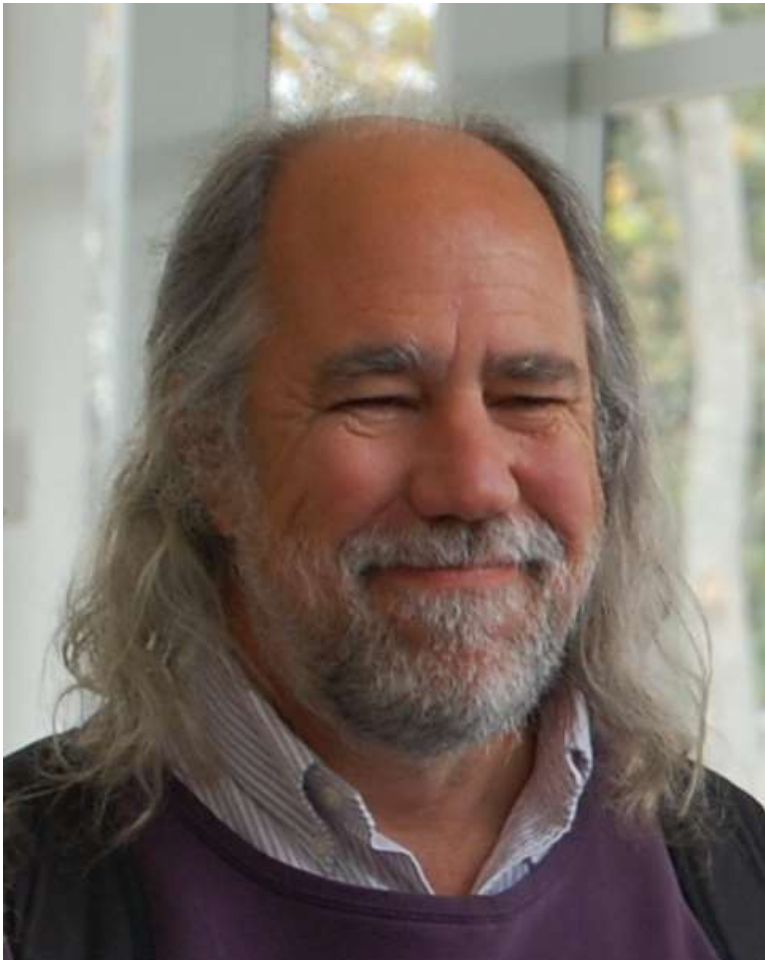
Sofrem pressão para cumprir prazos e orçamentos apertados



Ficam inseguros com as mudanças de tecnologia que afetam sua qualificação em relação ao mercado



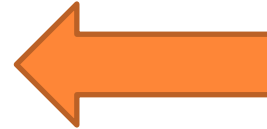
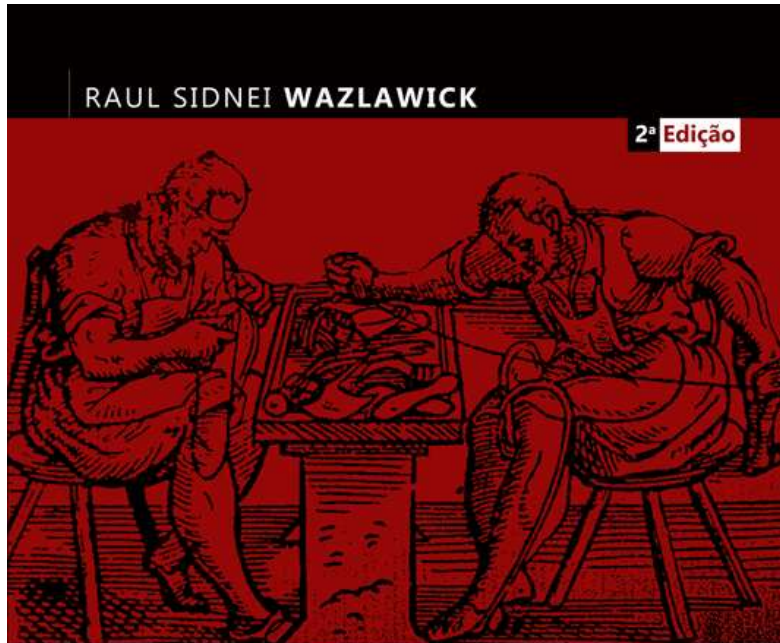
# GRADY BOOCH



- “Uma doença que dure tanto tempo quanto esta, francamente, deveria ser chamada de normalidade”
- Está mais para uma “aflição crônica”.



# AINDA FAZEMOS SOFTWARE COMO SE FAZIA SAPATOS NA IDADE MÉDIA?



## Engenharia de Software



CONCEITOS E PRÁTICAS



# OS ETERNOS MITOS



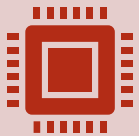
- Administrativos
- Do cliente
- Do profissional



# MITOS ADMINISTRATIVOS



*A existência de um manual de procedimentos e padrões é suficiente para a equipe produzir com qualidade.*



*A empresa deve produzir com qualidade, pois tem ferramentas e computadores de última geração.*



*Se o projeto estiver atrasado sempre é possível adicionar mais programadores para cumprir o cronograma.*



# MITOS DO CLIENTE



Uma declaração geral de objetivos é suficiente para iniciar a fase de programação. Os detalhes podem ser adicionados depois.



Os requisitos mudam com frequência, mas sempre é possível acomodá-los, pois o software é flexível.



Eu sei do que preciso.



# MITOS DO PROFISSIONAL



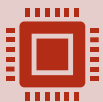
Assim que o programa for colocado em operação nosso trabalho terminou.



Enquanto o programa não estiver funcionando, não será possível avaliar sua qualidade.



Se eu esquecer alguma coisa, posso arrumar depois.



A única entrega importante em um projeto de software é o software funcionando.



# MITOS DE CONFIABILIDADE



O teste do software ou sua verificação formal pode remover todos os erros.



Aumentar a confiabilidade do software aumenta a segurança.



O reuso de software aumenta a segurança.





## CONCLUSÃO

**De nada adianta apenas estar consciente dos mitos. Para produzir software com mais qualidade e confiabilidade é necessário utilizar um série de práticas, algumas das quais serão apresentadas nesta disciplina.**

# (IN)DEFINIÇÃO DE ENGENHARIA DE SOFTWARE



- “A Engenharia de Software forma um aglomerado de conceitos que dizem absolutamente nada e que geram no estudante desta área um sentimento de ‘Nossa, li 15 kg de livros desta matéria e não aprendi nada. É tudo bom senso.’”
- Desciclopedia, 2010.





# UMA BOA DEFINIÇÃO

- Engenharia de Software é o processo de estudar, criar e otimizar os processos de trabalho para os desenvolvedores de software



# VAMOS APRENDER:



**Processos** de engenharia de software são **diferentes** dependendo do **tipo de software** que se vai desenvolver.



Dependendo do **nível de conhecimento** ou **estabilidade dos requisitos** deve-se optar por um ou outro ciclo de vida.



Uma área aparentemente tão subjetiva como “**riscos**” pode ser sistematizada e tratada efetivamente como um **processo de engenharia**.



Existem formas objetivas e padronizadas para **mensurar o esforço** do desenvolvimento de software.



# O ENGENHEIRO DE SOFTWARE

- Viabiliza e acompanha o processo de produção, fornecendo e avaliando as ferramentas e técnicas que julgar mais adequadas a cada projeto ou empresa.



# DIFERENTES PAPEIS



- Engenheiro de Software
- Gerente de Projetos
- Analista
- Projetista
- Programador



# EVOLUÇÃO DA ENGENHARIA DE SOFTWARE



- 1940 – Não havia software.
- 1950 – Início da computação comercial.
- 1960 a 1980 – Crise do software (mitos).
- 1980 a 1990 – Balas de prata.
- 2000 – Métodos ágeis.
- 2004 – Swebok.
- 2014 – Swebok 3.0.



# ÁREAS DO CONHECIMENTO DO SWEBOK 3.0

- Requisitos de Software
- Projeto de Software
- Construção de Software
- Teste de Software
- Manutenção de software
- Gerência de Configuração de Software
- Gerência da Engenharia de Software
- Processos de Engenharia de Software
- Ferramentas e Métodos da Engenharia de Software
- Qualidade de Software
- Práticas Profissionais em Engenharia de Software
- Economia da Engenharia de Software
- Fundamentos de Computação
- Fundamentos de Matemática
- Fundamentos de Engenharia



Cobertas neste livro





# TIPOS DE SOFTWARE DO PONTO DE VISTA DA ENGENHARIA

---

Software básico

---

Software de tempo real

---

Software comercial

---

Software científico e de engenharia

---

Software embutido e embarcado

---

Software pessoal (aplicativos)

---

Jogos

---

Inteligência artificial

---

...

