

Engenharia de Software I

OBJETIVOS, CONCEITOS E EVOLUÇÃO

Prof. Claudemir Santos Pinto

profdemir@yahoo.com.br

Pra começar... o que é software?

Pressman (1997) diz que um livro didático poderia trazer a descrição de um software da seguinte forma:

- Instruções (programas de computador) que, quando executadas, produzem a função e desempenho desejados;
- Estruturas de dados que possibilitam que os programas manipulem adequadamente a informação;
- Documentos que descrevem a operação e o uso dos programas;

Pra começar... o que é software?

Sommerville (2003) complementa criando dois tipos diferentes:

- Produtos Genéricos (desenvolvidos para uma variedade de clientes)
- Produtos sob encomenda (para um único cliente, de acordo com suas especificações)

Pode-se dizer então, que software é: produto (ou serviço) que possui um grande número de características próprias oriundas dos critérios de análise, codificação e documentação aplicados durante sua construção.

O que você acha que é ?



Falaí !!!

Curiosidades sobre software...

<https://informationisbeautiful.net/visualizations/million-lines-of-code/>

<https://megacuriosidades.net/software-esta-comendo-o-mundo/>

Como tudo começou...

Evolução do hardware
X
Evolução do software



Um pouco de história: a crise de 1968

- Necessidade de softwares mais complexos com:

- Custos maiores;
- Baixa confiabilidade;
- Difícil manutenção; e
- Desempenho abaixo do esperado.

Percebeu-se então que uma abordagem informal de desenvolvimento não era suficiente.

Prováveis causas da crise: **falta de metodologia**

- as equipes de trabalho não tinham um modelo de como desenvolver
- não havia documentação adequada do que estava sendo executado
- “E agora: como dar manutenção em um sistema que não tem projeto?”

Mitos que contribuíram para a crise

1

A equipe possui
ferramentas de
desenvolvimento de
última geração!
Afinal, lhes
compramos os mais
novos computadores

Mais importante do que
ter um hardware de
última geração é ter
ferramentas para a
automatização do
desenvolvimento de
software (ferramentas
CASE, por exemplo)...

Mitos que contribuíram para a crise

2

Se os prazos atrasam, pode-se adicionar mais pessoas ao grupo e tirar o atraso

A introdução de novos profissionais numa equipe em fase de condução de um projeto vai requerer uma etapa de treinamento dos novos elementos da equipe; para isto, serão utilizados elementos que estão envolvidos diretamente no desenvolvimento, o que vai, consequentemente, implicar em maiores atrasos no cronograma.

Mitos que contribuíram para a crise

3

Uma descrição geral dos objetivos é suficiente para se escrever programas e os detalhes são vistos posteriormente

Este é um dos problemas que podem conduzir um projeto ao fracasso, o cliente deve procurar definir o mais precisamente possível todos os requisitos importantes para o software: funções, desempenho, interfaces, restrições de projeto e critérios de validação são alguns dos pontos determinantes do sucesso de um projeto.

Mitos que contribuíram para a crise

4

Os requisitos de projeto modificam-se continuamente, mas as mudanças podem ser facilmente acomodadas, porque o software é flexível

É verdade que o software é flexível. Entretanto, não existe software, por mais flexível que seja, que suporte alterações de requisitos significativas sem custo adicional de desenvolvimento. O fator de multiplicação nos custos de desenvolvimento do software devido a alterações nos requisitos cresce em função do estágio de evolução do projeto.

Mitos que contribuíram para a crise

5

Assim que escrevermos o programa e o colocarmos em funcionamento nosso trabalho estará completo.

O que ocorre na realidade é completamente diferente disto. Segundo dados obtidos a partir de experiências anteriores, 50 a 70% do esforço de desenvolvimento de um software é gasto após a sua entrega ao cliente (manutenção).

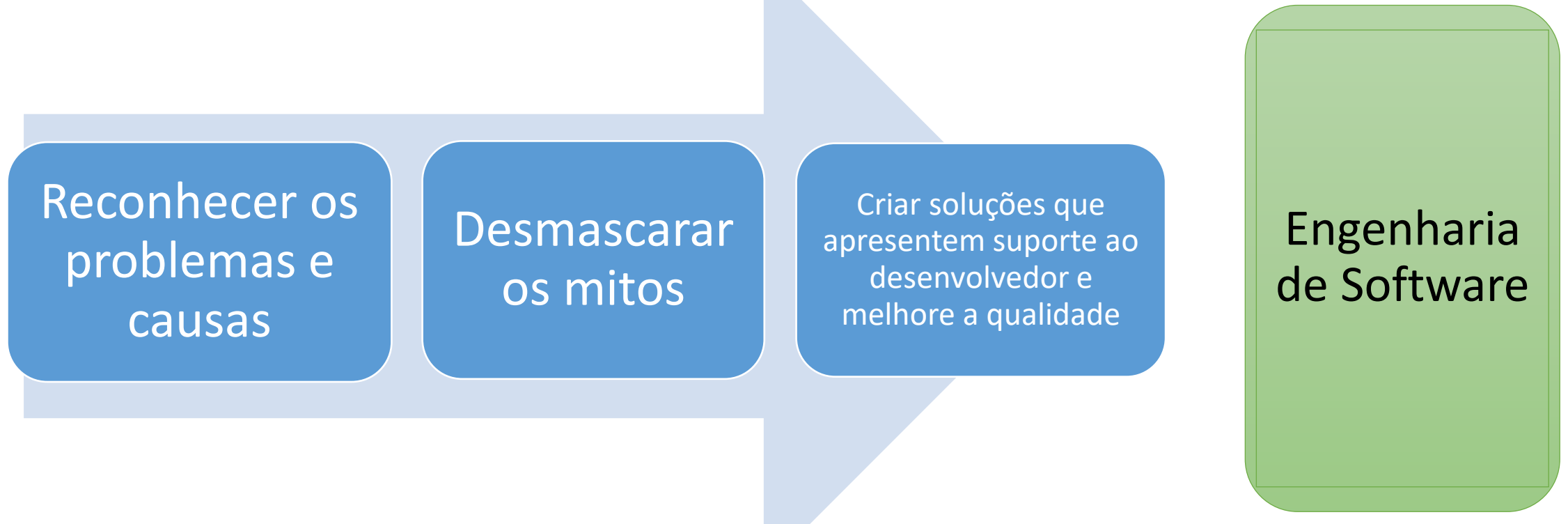
Mitos que contribuíram para a crise

6

Enquanto não tiver o programa “funcionando” eu não terei realmente nenhuma maneira de avaliar sua qualidade

Na realidade, a preocupação com a garantia do software deve fazer parte de todas as etapas do desenvolvimento, sendo que, ao fim de cada uma destas etapas, os documentos de projeto devem ser revisados observando critérios de qualidade.

Possíveis soluções:



A ideia então foi incluir a

Engenharia é aplicação do conhecimento científico, econômico, social e prático, com o intuito de inventar, desenhar, construir, manter e melhorar estruturas, máquinas, aparelhos, sistemas, materiais e processos.

Definições de Engenharia de Software:

Segundo Boehm(1976)

- A aplicação prática do conhecimento científico para o projeto e a construção de programas computacionais e a documentação necessária à sua operação e manutenção.

Segundo Krakowiak(1985)

- Conjunto de métodos, técnicas e ferramentas necessárias à produção de software de qualidade para todas as etapas do ciclo de vida do produto.

Definições de Engenharia de Software:

Segundo IEEE(1992)

- Engenharia de software é a aplicação de uma abordagem sistemática, disciplinada e quantificável, para o desenvolvimento, operação e manutenção do software.

Segundo Sommerville (2003)

- Disciplina que trabalha todos os aspectos da produção de software, desde os estágios iniciais de especificação até a manutenção deste, depois que ele entrou em operação.

Definições de Engenharia de Software:

De modo mais objetivo, pode-se dizer que a Engenharia de Software busca prover a tecnologia necessária para produzir software de alta qualidade a um baixo custo. Os dois fatores motivadores são essencialmente a **qualidade** e o **custo**.



A Engenharia de Software:

Não se dedica apenas aos processos técnicos de software, mas também:

- Ao gerenciamento dos projetos;
- Ao desenvolvimento de ferramentas, métodos e teorias que dêem apoio à produção de software e favoreçam o trabalho em equipe.

Abrangência:



Abrangência:



Há diferentes métodos para as diferentes etapas do desenvolvimento

Ferramentas para análise, gerenciadores de banco de dados, linguagens de programação, ferramentas case, etc

Define a sequência dos métodos que serão aplicados e as ferramentas que serão disponibilizadas.

Fundamentos da Engenharia de Software

Considera-se que a Engenharia de Software é uma área interdisciplinar, baseada nos fundamentos de:

- Ciências da computação
- Administração de projetos
- Comunicação
- Técnicas de solução de problemas

Princípios da Engenharia de Software

Além dos fundamentos, alguns princípios ou elementos predominantes deram origem à engenharia de software. Alguns podem ser destacados:

- Formalidade para evitar a dependência de determinadas pessoas ou processos
- Abstração para identificar os aspectos mais importantes de determinado fenômeno
- Decomposição para subdividir problemas complexos
- Generalização para disseminar soluções semelhantes e reutilizar resultados

Desafios da Engenharia de Software

- As economias de todas as nações desenvolvidas são dependentes de software.
- Cada vez mais sistemas são controlados por software.
- Os investimentos em software representam uma fração significativa do PIB em todos os países desenvolvidos.

Responda com suas palavras:

- Como a Engenharia de Software pode influenciar o trabalho das equipes de desenvolvimento?
- Cite e explique 2 mitos que influenciaram na Crise de 1968 e que podem afetar o desenvolvimento de software nos dias atuais.
- Quais os 2 fatores que motivam a utilização da Engenharia de Software na produção de software? Explique.

Enviar para: profdemir@yahoo.com.br

Com o assunto: **ES1 – engenharia**

Prazo: 18/02/2020