



**NÃO ME
PERGUNTE
SE SOU CAPAZ
APENAS ME DÊ
A MISSÃO**

VICTOR GREEN



ENGENHARIA DE SOFTWARE

Introdução

Professor: Charles Leite

Contextualização

Engenharia de software

- As economias de TODAS as nações desenvolvidas são dependentes de software.
- Cada vez mais sistemas são controlados por software.
- A engenharia de software se dedica às **teorias, métodos e ferramentas** para desenvolvimento de software profissional
 - Sistemas **não-triviais**
 - Com base em um conjunto de **requisitos**
 -

Custos de software

- Os custos de software **dominam os custos** de sistemas computacionais.
 - Em geral, software custa mais que hardware
- Manter um software custa mais que desenvolvê-lo
- A engenharia de software dedica-se ao desenvolvimento de software com custos adequados
 - **Respeitando** o cronograma acordado
 - **Satisfazendo** as necessidades dos clientes
 - **Minimizando** o custo de manutenção

FAQs sobre engenharia de software

- O que é software?
- O que é engenharia de software?
- Qual é a diferença entre engenharia de software e ciência da computação?
- Qual é a diferença entre engenharia de software e engenharia de sistemas?
- O que é processo de software?
- O que é um modelo de processo de software?

FAQs sobre engenharia de software

- Quais são os custos da engenharia de software?
- O que é CASE (*Computer-Aided Software Engineering*)
- Quais são os atributos de um bom software?
- Quais são os desafios-chave enfrentados pela engenharia de software?

O que é software?

- Programas de computador e artefatos associados
- Produtos de software podem ser:
 - **Genéricos** – desenvolvidos para serem vendidos para uma grande variedade de clientes (e.g., Excel e Word)
 - **Personalizados** – desenvolvidos para um único cliente de acordo com as suas especificações.

O que é software?

- Um software novo pode ser criado através de:
 - Desenvolvimento de novos programas;
 - Configuração de sistemas de software genéricos;
 - Reutilização de um software existente.
 -

Classificação de Software



- Aplicativos
- Sistemas Embarcados
- Sistemas Móveis
- Driver
- Sistemas Operacionais
-

O que é engenharia de software?

Engenharia de software é uma disciplina relacionada **com todos os aspectos da produção** de software e propõe ferramentas, técnicas e processos para:

- Entender com precisão qual é o problema (as necessidades associadas ao sistema que deve ser **construído/modificado**)
- Produzir **uma solução adequada** para esse problema (um sistema pronto para usar, levando-se em consideração as necessidades das partes interessadas)
- Levando-se em conta **restrições de desenvolvimento** e **recursos disponíveis**

Qual é a diferença entre engenharia de software e ciência da computação?

- A ciência da computação dedica-se à **teoria** e aos **fundamentos**;
 - Engenharia de software dedica-se aos **aspectos práticos de desenvolvimento e de entrega de software**
 - “**O que**” vs. “**Como**”
- Teorias de ciência da computação são ainda insuficientes para atuar como uma base completa para a engenharia de software.
 - Em outras palavras: **não há receitas prontas!**

Qual é a diferença entre engenharia de software e engenharia de sistemas?

- Engenharia de sistemas:
 - Mais ampla
 - Muita ênfase em aspectos de hardware e infra-estrutura
 - Abstração do hardware
 - Organização física das partes do sistema
 - Aspectos de comunicação
 - **Engloba** a engenharia do software
- Os engenheiros de sistema estão envolvidos em diversas atividades da engenharia de software
 - Projeto da arquitetura
 - Elicitação e especificação de requisitos

O que é processo de software?

- Um conjunto estruturado de atividades, práticas, artefatos e ferramentas necessários para o desenvolvimento de um sistema de software
 - Especificação;
 - Projeto;
 - Validação;
 - Evolução.

O que é processo de software?

- Alguns elementos de um processo:
 - **Modelos de sistema:**
 - Modelos gráficos que podem/devem ser produzidos e as notações que devem ser empregadas;
 - Restrições aplicadas aos modelos de sistema;
 - **Recomendações** de boas práticas de projeto;
 - **Atividades** que devem ser seguidas em determinada ordem
 - Às vezes também prescrevem **ferramentas**
- Um processo adere a um ou mais **modelos de processo**

.

O que é um modelo de processo de software?

- **Uma representação simplificada** de um processo de software, apresentado sob uma perspectiva específica
 - Incluindo algumas atividades e sua organização de alto nível
- Modelos gerais de processo
 - Cascata;
 - Desenvolvimento iterativo;
 - Engenharia de software baseada em componentes.
- Representações de modelos de processo:
 - Modelo de *workflow* – sequência de atividades;
 - Modelo de fluxo de dados – fluxo de informações;
 - Modelo de papel/ação – quem faz o quê.
 -

Quais são os custos da engenharia de software?

- 60% são custos de **desenvolvimento**
- 40% são custos de **testes**
- Para software sob encomenda, os custos de evolução normalmente **excedem** os de desenvolvimento.
- Os custos variam dependendo do tipo de sistema que está sendo desenvolvido e dos requisitos do sistema, tais como **desempenho** e **confiabilidade**

O que é CASE

(*Computer-Aided Software Engineering*)

- Sistemas de software que se destinam a fornecer apoio automatizado para as atividades de desenvolvimento de software.
- Sistemas CASE são usados frequentemente para apoiar um método específico
- **Upper-CASE**
 - Ferramentas para apoiar as atividades iniciais do processo de requisitos e de projeto;
- **Lower-CASE**
 - Ferramentas para apoiar as atividades finais tais como programação, *debugging* e teste.

Quais são os atributos de um bom software?

- O software deve fornecer a **funcionalidade e o desempenho** requeridos para o usuário e deve apresentar certas características
- **Facilidade de manutenção**
 - Deve ser fácil e barato fazer com que o sistema, depois de implantado, evolua para atender às necessidades dos clientes
- **Confiabilidade**
 - O software deve funcionar conforme sua especificação
- **Eficiência**
 - O software deve usar o mínimo de recursos e prover máxima funcionalidade
- **Usabilidade**
 - O software deve ser compreensível e fácil de usar

Quais são os desafios-chave enfrentados pela engenharia de software?

- **Heterogeneidade**

- Sistemas de software devem ser capazes de lidar com diferentes plataformas de hardware e ambientes de execução;

- **Entrega**

- O sistema deve ser entregue ao cliente no menor tempo possível, com o menor custo possível;

- **Confiança**

- O usuário deve poder justificadamente depositar sua confiança no sistema

- **Escala**

- O sistema deve funcionar adequadamente mesmo quando um grande número de usuários o está usando

Responsabilidade profissional e ética

- A engenharia de software envolve responsabilidades mais amplas do que simplesmente a aplicação de habilidades técnicas.
- Os engenheiros de software devem se comportar de modo honesto e eticamente responsável para serem respeitados como profissionais.
- O comportamento ético é mais do que simplesmente a sustentação de leis.

Questões de responsabilidade profissional

- **Confidencialidade**

- Os engenheiros de software devem normalmente respeitar a confidencialidade de seus funcionários ou clientes, independentemente de ter ou não assinado um acordo formal.
- Caso não aceitem essas condições, devem deixar isso explícito para seus contratantes

- **Competência**

- Os engenheiros não devem conscientemente aceitar um trabalho que esteja fora de sua competência.

Questões de responsabilidade profissional

- Direitos sobre propriedade intelectual
 - Desenvolvedores devem estar cientes das leis locais que regem o uso de propriedade intelectual, tais como patentes, direitos autorais, etc.
 - Eles devem tomar cuidado para assegurar que a propriedade intelectual dos funcionários e clientes seja protegida.

Dilemas éticos

- Discordância, em princípio, das políticas da gerência sênior.
- Um funcionário age de uma forma não ética e libera um sistema de segurança crítico sem finalizar o teste do sistema.
- Participação no desenvolvimento de sistemas de **armamentos militares** ou de **sistemas nucleares**

DÚVIDAS ...



Referências

- SOMMERVILLE, Ian. **Engenharia de Software**. *Capítulo 1, páginas 02-17*. 9ª ed. – São Paulo: Pearson Prentice Hall, 2011.