

Engenharia de Software

Capítulo 1 - Introdução

Prof. Nilson Júnior

<http://professornilson.wix.com/ifpe>

nilson.junior@jaboatao.ifpe.edu.br

Slides originais elaborados por Ian Sommerville
O autor permite o uso e a modificação dos *slides* para fins didáticos

Agenda do semestre

- ▶ **Introdução** • Visão geral da área de engenharia de software; • Conceitos de produto (sistemas de software) e processo de desenvolvimento de software; • Paradigmas (modelos de processo).
- ▶ **Gerenciamento do desenvolvimento de projetos de software**; • Definição de Metas e objetivos; • Estimativas: métricas e modelos de custo; • Estudo de viabilidade • Cronogramas.
- ▶ **Extração de requisitos de sistemas de software** • Conceitos; • Técnicas para extração de requisitos;
- ▶ **Análise de requisitos de sistemas de software**; • Modelos do sistema: funcional, comportamental, de dados e baseado em objetos; • Documentação da especificação do sistema.
- ▶ **Projeto de Sistemas de Software** • Arquitetura de sistemas; • Projeto detalhado de sistemas; • Qualidade do projeto: coesão, acoplamento e outros; • Documentação do projeto.
- ▶ **Estudo de Caso** • Extração de requisitos de um sistema de software; • Análise do sistema utilizando os modelos estudados; • Projeto da arquitetura do sistema; • Implementação do sistema: construção de um protótipo
(PROJETO E PRÁTICA II)

Tópicos abordados

- ▶ Introdução sobre engenharia de software.
- ▶ Responsabilidade profissional e ética.

Avaliação

Nota Unidade 1 = Avaliação Escrita +
(Exercícios e Trabalhos)

Nota Unidade 2 = Avaliação Escrita +
(Exercícios e Trabalhos)

Média = ((Nota Unidade 1 + Nota Unidade
2) / 2)

Cenário Atual

- ▶ As economias de TODAS as nações desenvolvidas são dependentes de software.
- ▶ Cada vez mais os sistemas são controlados por software.
- ▶ A engenharia de software se preocupa com teorias, métodos e ferramentas para o desenvolvimento profissional de software.

Custos do Software

- ▶ Os custos do software geralmente dominam os custos do sistema total.
 - ▶ Os custos de software em um PC são geralmente maiores que o custo do hardware
- ▶ Softwares são mais caros para manter do que para desenvolver.
 - ▶ Para sistemas com uma vida longa, os custos com manutenção podem ser muitas vezes maiores que os custos de desenvolvimento.
- ▶ Engenharia de software preocupa-se com o desenvolvimento de softwares rentáveis.

FAQs sobre engenharia de software

O que é software?

O que é engenharia de software?

Qual a diferença entre engenharia de software e Ciência da Computação?

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

O que é um processo de desenvolvimento de software?

O que é um modelo de desenvolvimento de software?

FAQs sobre engenharia de software

Quais os custos da engenharia de software?.

Quais os métodos de engenharia de software?

O que é CASE (Computer-Aided Software Engineering).

Quais os atributos de um bom software?

Quais os principais desafios enfrentados pela engenharia de software?.

O que é software?

- ▶ Programas de computador e documentação associada.
- ▶ Produtos de Software podem ser desenvolvidos para um cliente em particular ou para um mercado geral.
- ▶ Produtos de Software podem ser
 - ▶ **Genéricos** - desenvolvidos para serem vendidos para uma rede de clientes diferentes.
 - ▶ **Sob medida** (personalizados) - desenvolvidos para um único cliente de acordo com suas especificações.

O que é Engenharia de Software?

- ▶ **Engenharia de software** é uma disciplina da engenharia que se preocupa com todos aspectos da produção de software.
- ▶ **Engenheiros de software** devem adotar uma postura sistemática e organizada para seus trabalhos e utilizar ferramentas e técnicas apropriadas, dependendo do problema a ser resolvido, dos obstáculos no desenvolvimento e dos recursos disponíveis.

Qual a diferença entre engenharia de software e Ciência da Computação?

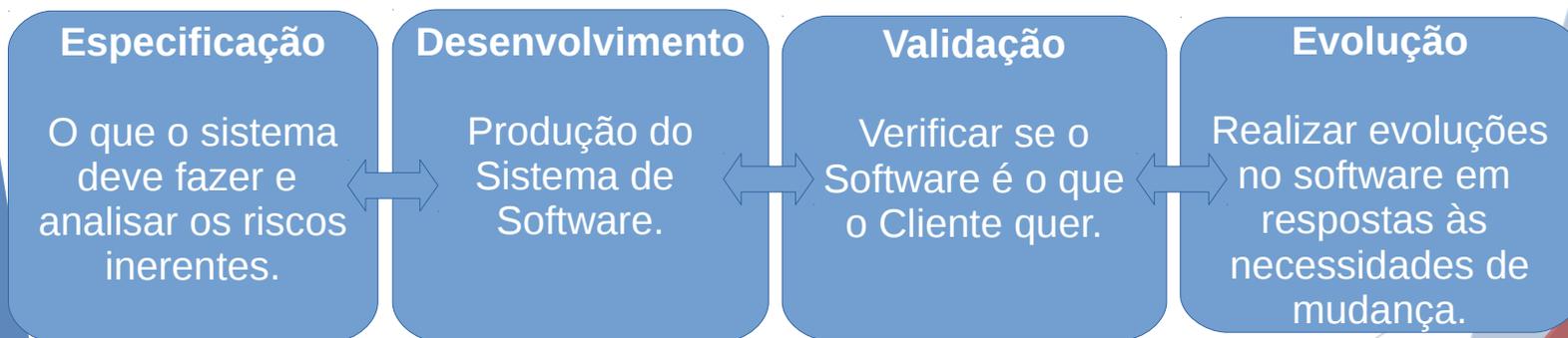
- ▶ **A Ciência da Computação** preocupa-se com teorias e fundamentos; a engenharia de software está preocupada com a viabilidade de desenvolvimento e entrega de softwares úteis.

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

- ▶ **Engenharia de sistema** preocupa-se com todos os aspectos do desenvolvimento de sistemas baseados em computadores, incluindo hardware, software e engenharia de processo. Engenharia de software é parte deste processo.
- ▶ **Os engenheiros de sistema** envolvem-se na especificação do sistema, no design arquitetural, na integração e na implantação.

O que é um processo de desenvolvimento de software?

O processo de software é um conjunto de atividades destinadas à produção e/ou evolução de um sistema. As atividades dos processos podem envolver o desenvolvimento ou a manutenção do software através de linguagens de programação como php ou Java. As atividades genéricas em todos os processos de software são:



O que é um modelo de desenvolvimento de software?

“É uma representação simplificada de um processo de software, apresentada de uma perspectiva específica”.

Exemplos de perspectivas de processo são:

- Perspectiva de fluxo de trabalho - sequência de atividades.
- Perspectiva de fluxo de dados - fluxo de informação.
- Perspectiva de papel/ação - quem faz o quê.

Modelos de Processos

- Cascata.
- Desenvolvimento Evolucionário.
- Modelos Ágeis.
- Modelo Espiral.
- Iterativo e Incremental.
- Integração de Componentes Reutilizáveis.

Elementos da Engenharia de Software

1. Coloque em uma panela funda o leite condensado, a margarina e o chocolate em pó.
2. Cozinhe [no fogão] em fogo médio e mexa sem parar com uma colher de pau.
3. Cozinhe até que o brigadeiro comece a desgrudar da panela.
4. Deixe esfriar bem, então unte as mãos com margarina, faça as bolinhas e envolva-as em chocolate granulado.

Elementos da Engenharia de Software

O que é processo, método ou ferramenta?

1. **Coloque** em uma **panela** funda o leite condensado, a margarina e o chocolate em pó.
2. **Cozinhe** [no **fogão**] em fogo médio e **mexa** sem parar com uma **colher de pau**.
3. **Cozinhe** até que o brigadeiro comece a desgrudar da **panela**.
4. Deixe esfriar bem, então **unte** as **mãos** com margarina, **faça** as **bolinhas** e **envolva**-as em chocolate granulado.

Processo

 método

 ferramenta

Quais os custos da engenharia de software?

- Cerca de 60% dos custos são de desenvolvimento, 40% são custos de teste. Para software personalizado, os custos de evolução geralmente excedem os custos de desenvolvimento.
- Os custos variam dependendo do tipo de sistema a ser desenvolvido e dos requisitos dos atributos de sistema como performance e confiabilidade do sistema.
- Distribuição de custos depende do modelo de desenvolvimento que é usado.

O que é CASE (Computer-Aided Software Engineering)

- Sistemas de Software que devem prover um suporte automatizado para atividades do processo de software. Sistemas CASE são geralmente utilizados para suporte de método.

CASE de alto baixo (UpperCase)

Ferramentas para dar suporte às atividades iniciais de requisito e projeto.

CASE de nível baixo (LowerCase)

Ferramentas para dar suporte às atividades posteriores como programar, depurar e testar.

Quais os atributos de um bom software?

- O software deve fornecer as funcionalidades e performance requeridas para o usuário e deve ser fácil de manter, confiável e utilizável

Manutenibilidade

O software deve evoluir para atender às necessidades de mudança

Confiabilidade

O software deve ser confiável

Eficiência

O software não deve fazer uso desnecessário de recursos do sistema

Usabilidade

O software deve ser utilizável pelos usuários para os quais ele foi projetado

Quais os principais desafios enfrentados pela engenharia de software?

- Lidando com sistemas legados, lidando com a diversidade crescente e lidando com a demanda de tempos para entrega reduzidos.

Sistemas legados

Sistemas antigos e de valor devem ser mantidos e atualizados.

Heterogeneidade

Os sistemas são distribuídos e incluem um misto de hardware e software.

Entrega

Existe uma pressão crescente para agilizar a entrega de software.

Mitos e Realidades - Gerenciamento

- ▶ **Mito.** "Se a equipe dispõe de um manual repleto de padrões e procedimentos de desenvolvimento de software, então a equipe está apta a encaminhar bem o desenvolvimento."
- ▶ **Realidade.** Isto verdadeiramente não é o suficiente... é preciso que a equipe aplique efetivamente os conhecimentos apresentados no manual... é necessário que o que conste no dado manual reflita a moderna prática de desenvolvimento de software.

Mitos e Realidades - Gerenciamento

- ▶ **Mito.** "Se o desenvolvimento do software estiver atrasado, basta aumentar a equipe para honrar o prazo de desenvolvimento."
- ▶ **Realidade.** Isto também dificilmente vai ocorrer na realidade, alguém disse um dia que "acrescentar pessoas em um projeto atrasado vai torná-lo ainda mais atrasado". De fato, 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, conseqüentemente, implicar em maiores atrasos no cronograma.

Mitos e Realidades - Cliente

- ▶ **Mito.** "Uma descrição breve e geral dos requisitos do software é o suficiente para iniciar o seu projeto... maiores detalhes podem ser definidos posteriormente."
- ▶ **Realidade.** 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.

Responsabilidade ética e profissional

- A engenharia de software envolve responsabilidades maiores do que simplesmente a aplicação de habilidades técnicas.
- Engenheiros de software devem comportar-se de uma forma honesta e eticamente responsável se quiserem ser respeitados como profissionais.
- Comportamento ético é mais que simplesmente manter a lei.

Questões de responsabilidade profissional

Confiabilidade

Engenheiros devem normalmente respeitar a confidencialidade de seus patrões ou clientes independente de ter ou não um acordo formal de confiabilidade assinado.

Competência

Engenheiros não devem falsear seus níveis de competência. Não devem deliberadamente aceitar trabalhos que estejam fora de suas competências.

Questões de responsabilidade profissional

Direitos de propriedade intelectual

Engenheiros de software devem conhecer as regras locais que governam o uso de propriedade intelectual como as patentes, copyright, etc. Eles devem ser cuidadosos para garantir que a propriedade intelectual de seus patrões e clientes esteja protegida.

Mau-uso do Computador

Engenheiros de software não devem usar suas habilidades técnicas para fazer mau uso do computador de outras pessoas. O mau uso do computador vai de relativamente trivial (brincar com jogos na máquina do patrão, vamos dizer assim) a extremamente sérios (disseminação de vírus).

Código de ética - princípios

1. PÚBLICO

Engenheiros de software devem agir em consistência com o interesse do público.

2. CLIENTE E PATRÃO

Engenheiros de software devem agir de modo a visar os melhores interesses de seus clientes e patrões e consistente com o interesse do público.

3. PRODUTO

Engenheiros de software devem assegurar que seus produtos e modificações relacionadas atendam aos mais altos padrões profissionais possíveis.

Código de ética - princípios

4. JULGAMENTO

Engenheiros de software devem manter a integridade e independência em seu julgamento profissional.

5. GERENCIAMENTO

Gerentes e líderes de engenharia de software devem assinar e promover uma abordagem ética em relação ao gerenciamento de desenvolvimento e manutenção de software.

6. PROFISSÃO

Engenheiros de software devem adiantar a integridade e reputação da profissão consistente ao interesse do público.

Código de ética - princípios

7. COLEGAS

Engenheiros de software devem ser justos e apoiar seus colegas.

8. PARA SI

Engenheiros de software devem participar de um aprendizado durante toda sua vida no que diz respeito à prática de sua profissão e devem promover uma abordagem ética à prática da profissão.

Conclusões

- A engenharia de software é uma disciplina da engenharia que se preocupa com todos os aspectos da produção de software.
- Produtos de software consistem em programas desenvolvidos e documentação associada. Atributos essenciais dos produtos são manutenibilidade, dependência, eficiência e usabilidade.
- O processo do software consiste em atividades que estão envolvidas no desenvolvimento de produtos de software. Atividades básicas são especificação, desenvolvimento, validação e evolução de software.
- Os métodos são modos organizados de produzir software. Eles incluem sugestões para o processo a serem seguidas, as notações a serem utilizadas, as regras que governam as descrições do sistema que são produzidas e o projeto de pautas.

Conclusões

- Ferramentas CASE são sistemas de software projetados para suportar atividades rotineiras no processo do software como editar diagramas de design, verificar consistência do diagrama e manter rastro de testes de programas que foram executados.
- Engenheiros de software têm responsabilidades para com a profissão de engenheiro bem como para com a sociedade. Eles não devem preocupar-se somente com assuntos técnicos.
- Sociedades profissionais publicam códigos de conduta que estabelecem os padrões de comportamento esperado de seus membros.