

Introdução

- Apresentar a engenharia de software

Objetivos

- Apresentar a engenharia de software e explicar a sua importância
- Responder as questões principais sobre engenharia de software
- Apresentar questões éticas e profissionais e explicar porquê elas são preocupações de engenheiros de software

Tópicos abordados

- FAQs sobre engenharia de software
- Responsabilidade profissional e ética

Engenharia de Software

- As economias de TODAS as nações desenvolvidas dependem de software
- Mais e mais sistemas são controlados por software
- Engenharia de software preocupa-se com as teorias, os métodos e as ferramentas para o desenvolvimento profissional de software
- O gasto com engenharia de software representa uma parte significativa do PIB em todos os países desenvolvidos

Custos de software

- Os custos de software frequentemente dominam os custos do sistema. Os custos com o software em um PC são frequentemente maiores que os custos com o hardware
- Os custos de software são maiores para mantê-lo do que para desenvolvê-lo. Para sistemas com vida longa, os custos de manutenção podem ser várias vezes maiores que os de desenvolvimento
- A engenharia de software preocupa-se com o desenvolvimento de software com um custo viável

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 um processo de software?
- O que é um modelo de processo de software?

FAQs sobre engenharia de software

- O que são custos de engenharia de software?
- O que são os métodos de engenharia de software?
- O que é CASE (Computer-Aided Software Engineering)
- Quais são os atributos de bons software?
- Quais são os principais desafios a serem enfrentados pela engenharia de software?

O que é software?

- Os programas de computador e a documentação associada
- Os produtos de software podem ser desenvolvidos para um cliente específico ou podem ser desenvolvidos para um mercado genérico
- Os produtos de software podem ser
 - Genérico – desenvolvido para ser vendido para uma gama de clientes diferentes
 - Encomendado (personalizado) – desenvolvido para um único cliente, de acordo com a sua especificação

O que é engenharia de software?

- Engenharia de software é uma área da engenharia que se preocupa com todos os aspectos da produção de software
- Engenheiros de software devem adotar uma abordagem sistemática e organizada para o seu trabalho e usar ferramentas e técnicas apropriadas, dependendo do problema a ser resolvido, das restrições de desenvolvimento e dos recursos disponíveis

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

- A ciência da computação preocupa-se com a teoria e os fundamentos; a engenharia de software preocupa-se com a prática do desenvolvimento e entrega de software útil
- As teorias da ciência da computação são atualmente insuficientes para servir como base única para a engenharia de software

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

- A engenharia de sistemas preocupa-se com todos os aspectos do desenvolvimento de sistemas baseados em computador, incluindo hardware, software e engenharia de processos. A engenharia de software faz parte desse processo
- Engenheiros de sistemas estão envolvidos na especificação, projeto arquitetural, integração e desenvolvimento de sistemas

O que é um processo de software?

- Um conjunto de atividades cujo objetivo é o desenvolvimento ou a evolução de software
- As atividades genéricas em todos os processos de software são:
 - Especificação – o que o sistema deve fazer e as restrições aplicadas ao seu desenvolvimento
 - Desenvolvimento – produção do sistema de software
 - Validação – verificar que o software é o que o cliente quer
 - Evolução – modificar o software em resposta a necessidades de alteração

O que é um modelo de processo de software?

- Uma representação simplificada de um processo de software, apresentado a partir de uma perspectiva específica
- Exemplos de perspectivas do processo são
 - Perspectiva de Fluxo de Trabalho – sequência de atividades
 - Perspectiva de Fluxo de Dados – fluxo de informações
 - Perspectiva Papel/Ação – quem faz o quê
- Modelos genéricos de processo
 - Cascata
 - Desenvolvimento evolucionário
 - Transformação formal
 - Integração de componentes reutilizáveis

©Ian Sommerville 2000

Software Engineering, 6th edition, Chapter 1

Slide 13

Quais são os custos da engenharia de software?

- Aproximadamente 60% dos custos são custos de desenvolvimento, e 40% são custos de testagem. Para softwares personalizados, os custos de evolução com frequência excedem os custos de desenvolvimento
- Os custos variam, dependendo do tipo de sistema que está sendo desenvolvido e dos requisitos de atributos de sistema, tais como desempenho e confiabilidade do sistema
- A distribuição de custos depende do modelo de desenvolvimento que está sendo usado

©Ian Sommerville 2000

Software Engineering, 6th edition, Chapter 1

Slide 14

O que são os métodos de engenharia de software?

- Abordagens estruturadas para o desenvolvimento de software
 - Incluem modelos do sistema, notações, regras, consultas de projeto e controle do processo
- Descrições do modelo
 - Descrições de modelos gráficos que devem ser reproduzidos
- Regras
 - Restrições aplicadas aos modelos do sistema
- Recomendações
 - Indicações de boas práticas de projeto
- Controle do processo
 - Quais atividades seguir

©Ian Sommerville 2000

Software Engineering, 6th edition, Chapter 1

Slide 15

O que é CASE (Computer-Aided Software Engineering)

- Sistemas de software cujo objetivo é fornecer apoio automatizado às atividades de processo de software. Sistemas CASE são frequentemente usados para apoio aos métodos
- Upper-CASE
 - Ferramentas que apoiam as atividades iniciais de requisitos e projeto
- Lower-CASE
 - Ferramentas que suportam atividades tardias, tais como programação, debugging e testes

©Ian Sommerville 2000

Software Engineering, 6th edition, Chapter 1

Slide 16

Quais são os atributos de um bom software?

- O software deve entregar a funcionalidade e desempenho exigidos pelo usuário e deve ser manutenível, digno de confiança e utilizável
- Manutenibilidade
 - O software deve evoluir para alcançar necessidades de mudança
- Confiabilidade
 - O software deve ser confiável
- Eficiência
 - O software não deve desperdiçar recursos do sistema
- Usabilidade
 - O software deve ser usável pelos usuários para os quais ele foi projetado

©Ian Sommerville 2000

Software Engineering, 6th edition, Chapter 1

Slide 17

Quais são os desafios principais a serem enfrentados pela engenharia de SW?

- Lidar com sistemas legados, lidar com diversidade crescente e lidar com necessidades de tempos de entrega reduzidos
- Sistemas legados
 - Sistemas velhos e valiosos devem ser mantidos e atualizados
- Heterogeneidade
 - Os sistemas são distribuídos e incluem uma combinação de hardware e software
- Entrega
 - Há uma pressão crescente para entrega mais rápida de software

©Ian Sommerville 2000

Software Engineering, 6th edition, Chapter 1

Slide 18

Responsabilidade profissional e ética

- A engenharia de software envolve responsabilidades mais abrangentes do que simplesmente a aplicação de habilidades técnicas
- Os engenheiros de software devem se comportar de uma forma honesta e eticamente responsável se eles pretendem ser respeitados como profissionais
- Comportamento ético significa mais do que simplesmente respeitar a lei

Questões de responsabilidade profissional

- *Confidencialidade*
 - Os engenheiros devem normalmente respeitar a confidencialidade de seus empregados ou clientes, não importando se um acordo de confidencialidade formal foi assinado.
- *Competência*
 - Os engenheiros não devem deturpar o seu nível de competência. Eles não devem aceitar trabalhos que reconhecidamente estão fora da sua competência.

Questões de responsabilidade profissional

- *Direitos de propriedade intelectual*
 - Os engenheiros devem estar cientes de leis locais que governam o uso de propriedade intelectual, tais como patentes, copyrights, etc. Eles devem ter o cuidado de garantir que a propriedade intelectual de empregados e clientes seja protegida.
- *Mau uso do computador*
 - Os engenheiros de software não devem usar suas habilidades técnicas para o mau uso de computadores de outras pessoas. O mau uso de computadores vai do relativamente trivial (jogar na máquina de uma empresa, por exemplo) a extremamente sério (disseminação de vírus).

Código de ética ACM/IEEE

- As sociedades profissionais dos EUA têm cooperado para produzir um código de ética prático
- Membros dessas organizações assinam o código de prática quando eles se associam
- O Código contém oito Princípios relacionados ao comportamento e a decisões tomadas por engenheiros de software profissionais, incluindo praticantes, educadores, gerentes, supervisores e planejadores, bem como estagiários e estudantes da profissão.

Código de ética – preâmbulo

- **Preâmbulo**
 - A versão curta do código resume aspirações em um alto nível de abstração; as cláusulas que são incluídas na versão completa dão exemplos e detalhes de como essas aspirações mudam a forma de como nós agimos como profissionais de engenharia de software. Sem as aspirações, os detalhes poderão parecer legalistas e tediosos; sem os detalhes, as aspirações podem se tornar grandiosas mas vazias; juntos, as aspirações e os detalhes formam um código coeso.
 - Os engenheiros de software devem se comprometer a tornar a análise, especificação, projeto, desenvolvimento, teste e manutenção de software uma profissão benéfica e respeitada. De acordo com o seu comprometimento com a saúde, segurança e bem-estar do público, os engenheiros de software devem aderir aos Oito Princípios a seguir:

Código de ética – princípios

- **1. PÚBLICO**
 - Engenheiros de software devem agir de forma consistente com o interesse público.
- **2. CLIENTE E EMPREGADO**
 - Engenheiros de software devem agir de uma maneira que esteja de acordo com os melhores interesses dos seus clientes e empregados, consistente com o interesse público.
- **3. PRODUTO**
 - Engenheiros de software devem garantir que os seus produtos e as modificações relacionadas a eles estejam de acordo com os mais altos padrões profissionais possíveis.

Código de ética – princípios

- 4. JULGAMENTO
 - Engenheiros de software devem manter integridade e independência em seu julgamento profissional.
- 5. GERENCIAMENTO
 - Gerentes e líderes de engenharia de software devem adotar e promover uma abordagem ética para o gerenciamento do desenvolvimento e da manutenção de software.
- 6. PROFISSÃO
 - Engenheiros de software devem fomentar a integridade e reputação da profissão, de forma consistente com o interesse público.

Código de ética – princípios

- 7. COLEGAS
 - Engenheiros de software devem ser justos e apoiadores de seus colegas.
- 8. PESSOAL
 - Engenheiros de software devem participar em um aprendizado constante envolvendo a prática da profissão e devem promover uma abordagem ética à prática da profissão.

Dilemas éticos

- Desacordo de princípios com as políticas do gerente senior
- O seu empregador age de uma forma não ética e lança um sistema de segurança crítica sem finalizar os testes do sistema
- Participação no desenvolvimento de sistemas de armas militares ou sistemas nucleares

Pontos principais

- Engenharia de software é uma área da engenharia que se preocupa com todos os aspectos da produção de software.
- Os produtos de software consistem nos programas desenvolvidos e a documentação a eles associada. Atributos essenciais do produto são a manutenibilidade, a confiabilidade, a eficiência e a usabilidade.
- O processo de software consiste de atividades que estão envolvidas no desenvolvimento de produtos de software. Atividades básicas são a especificação, o desenvolvimento, a validação e a evolução do software.
- Métodos são formas organizadas de se produzir software. Eles incluem sugestões do processo a ser seguido, as notações a serem usadas, as regras que controlam as descrições do sistema que são produzidas e linhas-guia de projeto.

Pontos principais

- Ferramentas CASE são sistemas de software que são projetadas para apoiar atividades de rotina no processo de software, tais como a construção de diagramas de projeto, a verificação da consistência dos diagramas e a manutenção do controle de testes do programa que tenham sido feitos.
- Engenheiros de software têm responsabilidades para a com a profissão e com a sociedade. Eles não devem apenas se preocupar com questões técnicas.
- As sociedades profissionais publicam códigos de conduta que definem os padrões de comportamento que se espera de seus membros.