

1 Aula 1

1.1 Introdução

1.1.1 Meta da Engenharia de Software

- Apoiar o desenvolvimento profissional de sistemas de software com boa relação custo-benefício

1.1.2 Importância

- A sociedade depende dos sistemas de software cada vez **mais avançados**
 - Mais avançados = mais complexos
- Evitar mudanças após o software começar a ser usado

1.1.3 Evolução da computação = evolução do software

1.1.4 Com o aparecimento de software mais complexo

- Abordagem informal de desenvolvimento não era o bastante
- Projetos com atrasos
- Custos mais elevados do que o previsto
- Dificil manutenção
- Resultado: Baixa qualidade

1.1.5 Crise do software = Problemas com:

- Maneira como o software era desenvolvido
- Maneira como era oferecido suporte a um volume crescente de SW
- Maneira como era suprida a demanda por mais softwares

1.1.6 Solução

- **Novas técnicas e métodos para controlar a complexidade inerente aos grandes sistemas de software**
- Da fase de concepção até a manutenção

1.1.7 Tripla restrição dos projetos

- Escopo
- Prazo
- Custo



1.2 Fundamentos da ES

1. Os softwares devem ser desenvolvidos em um processo gerenciado
 - O processo de desenvolvimento deve ser planejado
2. Confiança e desempenho são importantes para todos os tipos de softwares
3. Entender e gerenciar a especificação e os requisitos do software
4. Fazer o melhor uso possível dos recursos existentes

1.3 Perguntas frequentes

1. O que é software?
 - Programa + Documentação associada + Dados de configuração
 - Tipo de produtos de software
 - Produtos genéricos
 - Produtos sob encomenda
 - Qual é a diferença entre eles?
2. O que é a engenharia de software?
 - Disciplina que se ocupa de todos os aspectos de produção de software, desde os estágios iniciais de especificação até a sua manutenção
3. O que é o processo de desenvolvimento de software?
 - Conjunto de atividades e resultados que geram um produto de software
 - Atividade fundamentais:
 - Especificação do software: Definição do software a ser produzido e as restrições de suas operações
 - Desenvolvimento: O software é projetado e programado
 - Validação: O software é verificado para garantir que atenda o que o cliente quer
 - Evolução: O software é modificado para garantir o que o cliente quer
4. O que são modelos de processos de software?
 - Descrição de um processo, apresentado a partir de uma perspectiva específica.
5. Custos do processo de desenvolvimento?
 - A distribuição de custos ao longo do processo de desenvolvimento depende do processo utilizado e do tipo de software que está sendo desenvolvido. Aproximadamente 60% dos custos de software são de desenvolvimento e 40% são custos de testes.
6. O que são métodos?
 - Abordagens estruturadas para o desenvolvimento de software, cujo objetivo é facilitar a produção de software de qualidade
7. O que são ferramentas CASE - Computer-Aided Software Engineering?
 - Diferentes tipos de programas utilizados para apoiar as atividades de desenvolvimento de software
8. Quais os atributos de um bom software?
 - O conjunto de atributos depende da aplicação
 - Atender às necessidades do cliente



1.4 Principais desafios da Engenharia de Software

1.4.1 Características essenciais de um bom software

- Facilidade de manutenção
- Nível de confiança
- Eficiência
- Facilidade de uso
- Heterogeneidade
- Legado
- Tempo de entrega

1.5 Responsabilidade profissional e ética

1.5.1 O trabalho de engenharia de software é muito mais amplo do que aplicar habilidades técnicas

- Comportamento ético e moral para serem respeitados
- Comportamento ético é mais do que seguir leis
- Confidencialidade: independente de ter assinado acordo formal
- Competência: não deve aceitar trabalho consciente de que esteja fora de sua competência
- Direitos de propriedade intelectual: ter reconhecimento das leis e segui-las
- Má utilização de computadores: não usar suas habilidades técnicas para fazer mau uso de computadores de outras pessoas

1.6 Código de ética e de conduta profissional

1.6.1 ACM - Association for Computing Machinery

- <http://www.acm.org/about-acm/acm-code-of-ethics-and-professional-conduct>
- Início dos anos 90: pioneiro em mostrar pontos éticos na área
- Engloba todo tipo de profissional da área: deve ser respeitados por todos os membros
- Serve como base para a tomada de decisão ética na condução do trabalho profissional e para julgar violações de normas éticas profissionais
- Direitos de propriedade intelectual: ter conhecimento das leis e segui-las
- Composto por 24 imperativos

1.6.2 Imperativos gerais

1.6.3 Responsabilidades profissionais mais específicas

1.6.4 Imperativos de Liderança organizacional

1.6.5 Obediência ao Código

1.6.6 ACM/IEEE-CS (ACM e IEEE Computer Society)

- <http://www.acm.org/about/se-code>
- Final da década de 90



- Composto por 8 princípios

1. Público

- Agir de acordo com o interesse público
 - Aprovar software que contribua para o bem público
 - Alertar perigos associados ao software

2. Cliente e Empregador

- Agir para melhor interesse de seus clientes e empregadores
 - Não usar software pirata
 - Fornecer serviço de competência
 - Manter confiabilidade

3. Produto

- Assegurar que seus produtos e alterações tenham qualidade

4. Julgamento

- Assegurar integridade e independência em seu julgamento profissional
 - Não participar de práticas fraudulentas

5. Gestão

- Gerir o desenvolvimento de forma ética

6. Profissão

- Assegurar a integridade da profissão

7. Colegas

- Assegurar justiça e apoio aos colegas

8. Pessoal

- Manter constante prática de aprendizado e atualização nas práticas de sua profissão

1.7 Dilemas éticos

1.7.1 Discordância das políticas de alto escalão de sua empresa?

1.7.2 Seu empregador age de maneira não ética?

1.7.3 Participar do desenvolvimento de sistemas militares e nucleares?

1.7.4 Outros?