



Introdução a Engenharia de Software

Cleidson de Souza
cdesouza@ufpa.br
<http://www.ufpa.br/cdesouza>

Agenda

- Software e Engenharia de Software
- Engenharia de Software e Programação
- Histórico
 - **Crise do Software**
- “No Silver Bullet”
- Fases Genéricas do Processo de Desenvolvimento de Software
- Princípios, Métodos, Metodologias e Ferramentas de Engenharia de Software

2

Software

- Definição
 - **Programas (instruções) que executam uma determinada tarefa;**
 - **Estruturas de dados que permitem o armazenamento de informações;**
 - **Documentos que descrevem o uso do produto assim como o processo que levou a construção do mesmo.**
- Características
 - **Lógico (não se deteriora);**
 - **É construído de forma diferente do hardware.**

3

Engenharia de Software - Definições

- Engenharia de software é a disciplina que lida com a construção de sistemas de software flexíveis, modulares, robustos, confiáveis, usáveis e adequados ao contexto sócio-técnico onde estes sistemas estão inseridos.
- Uma disciplina que lida com a construção de sistemas de software que são tão grandes que são construídos por um ou vários times de engenheiros. [Ghezzi, Jazayeri, Mandrioli]
- Uma disciplina cujo objetivo é a produção de software sem falhas, entregue dentro de prazo e orçamento e satisfazendo as necessidades dos usuários. Além disso, o software precisa ser facilmente modificável quando as necessidades do usuário mudam.[Schach]

4

- Aplicação prática de conhecimento científico no projeto e construção de programas e da documentação requerida para desenvolver, operar e manter esses programas. [Boehm]
- O estabelecimento e uso de sólidos princípios de engenharia para que se possa obter economicamente software que seja confiável e que funcione eficientemente em máquinas reais. [Fritz Bauer-1969]
- Uma área do conhecimento voltada para a especificação, desenvolvimento e manutenção de sistemas de software aplicando tecnologias e práticas de ciência da computação, gerência de projetos e outras disciplinas, objetivando organização, produtividade e qualidade. [wikipedia]
- Construção de um software com várias versões por várias pessoas (Multi-person construction of multi-version software).[Parnas]

5

Engenharia e Programação

- | | |
|---|--|
| <ul style="list-style-type: none"> ■ Programação <ul style="list-style-type: none"> <input type="checkbox"/> Projeto pequeno <input type="checkbox"/> Você <input type="checkbox"/> Construir o que você mesmo quer <input type="checkbox"/> Um produto <input type="checkbox"/> Poucas modificações, feitas em seqüência <input type="checkbox"/> Tempo de vida curto <input type="checkbox"/> Barato <input type="checkbox"/> Pouco impacto | <ul style="list-style-type: none"> ■ Engenharia <ul style="list-style-type: none"> <input type="checkbox"/> Projeto Grande <input type="checkbox"/> Times <input type="checkbox"/> Construir o que os clientes querem <input type="checkbox"/> Família de produtos <input type="checkbox"/> Várias modificações, feitas em paralelo <input type="checkbox"/> Tempo de vida longo <input type="checkbox"/> Caro <input type="checkbox"/> Grande Impacto |
|---|--|



6

- Em resumo, engenharia de software inclui:
 - A organização de times para a construção colaborativa de sistemas de software
 - A determinação do que precisa ser construído;
 - Arquiteturas de software;
 - Análise e Testes;
 - Ciclos de vida e Processos de software;
 - Etc.

7

Windows Vista (Longhorn)

- 5000 desenvolvedores (sem incluir pessoal não-técnico);
- 40 milhões de linhas de código;
- 16 milhões de linhas de código somente nos últimos 3 anos;
- Versões compiláveis todos os dias;
- Testes de regressão;
- Intervalo de 3 dias para uma mudança submetida aparecer no executável;
- ~1.7 pessoas testando para cada programador;
- Precisa ter compatibilidade com versões anteriores;
- Precisa ser instalado em milhares de configurações diferentes;

8

Windows Vista (Longhorn)

- Como organizar as atividades tal que 5000 pessoas possam trabalhar juntas ao mesmo tempo;
 - **Processo de software;**
 - **Ferramentas de Gerência de Configuração;**
- Como testar tanto código? Para tantas plataformas diferentes?
 - **Testes de regressão?**
- Como projetar um sistema com 40 milhões de linhas de código? Como garantir a integridade deste projeto?
 - **Arquitetura / Projeto de software;**

9

WSSP

- ~36 estudantes de graduação;
- ~1 semestre;
- Sistema para a Polícia Militar - PA
 - **Cliente real: entrevistas, visitas, revisão de documentos;**
 - **190 e despacho de unidades integrado com sistemas de informação geográfica (SIGs);**
 - **Requisitos, Arquitetura flexível, Implementação, Testes, Manual, CVS, Rational ClearCase, etc.**

10

(Mais formalmente,) A Engenharia de Software inclui...

- Princípios, métodos e metodologias;
 - **Modularidade;**
 - **Diagramas de fluxo de dados;**
- Planejamento, estimativas, e gerência de projetos;
- Projeto de programas
 - **Algoritmos;**
 - **Estruturas de dados;**
- Ferramentas;
 - **Compiladores;**
 - **Geração de Testes;**
 - **Colaboração: email, CVS, Powerpoint;**

11

Histórico

- Termo cunhado durante uma conferência da OTAN em 1969 em Garmisch, Alemanha;
- O termo engenharia foi usado visando dar um sentido sistemático, rigoroso, formal, controlado para o desenvolvimento de software.
- Em resposta a chamada crise do software
 - **Termo usado para indicar o estado da prática em desenvolvimento de software na época;**

12

Crise do Software

- Constatações, que software(s)
 - Custos elevados;
 - Atrasos na entrega;
 - Baixa confiabilidade e corretude do produto;
 - Dificuldade de medição;
 - Gerência do desenvolvimento ineficaz;
 - Alta demanda não atendida;
 - Software existente é de difícil manutenção.
- Causas
 - Muita ênfase na programação;
 - Práticas “artesanal” utilizadas no desenvolvimento;
 - Pouco treinamento formal aos desenvolvedores.

13

No Silver Bullet ...

- Artigo de Fred Brooks publicado em 1986;
- Argumento
 - Nenhum ferramenta ou tecnologia irá produzir um aumento significativo (de ordens de magnitude) na produtividade do desenvolvimento de software
 - Porquê?
 - Complexidade Acidental
 - Erros de compilação, poder de expressividade das linguagens de programação, etc.
 - Complexidade Essencial
 - Complexidade, necessidade de modificações, Adaptabilidade, e Invisibilidade;

14

- Brooks discute porquê várias tecnologias não são a “silver bullet”
 - Ambientes de desenvolvimento integrado (IDEs);
 - Linguagem de programação Ada;
 - Orientação a Objetos;
 - Inteligência artificial;
 - Sistemas Especialistas;
 - Etc.
- Todas elas enfocam nas complexidade acidental (o açúcar sintático para o desenvolvimento de software);

15

Fases Genéricas do Processo de Desenvolvimento de Software

- Fase de Definição ou Análise: o quê
 - Engenheiro de Software tenta identificar: que informação deve ser processada, que função e desempenho são desejados, que comportamento deve ser esperado do sistema, que interfaces devem ser estabelecidas, quais restrições de projeto, quais critérios de validação
 - Os requisitos-chave do sistema e do software são identificados

16

■ Fase de Desenvolvimento (Projeto e Implementação): como

- **Definição de como os dados devem ser estruturados, como a função deve ser implementada dentro da arquitetura do software, como os detalhes procedimentais devem ser implementados, como as interfaces devem ser caracterizadas, como o projeto deve ser traduzido em linguagem de programação, e como o teste deve ser realizado;**

17

■ Fase de Manutenção: modificações

- **Tipos de Modificações: Corretiva, Adaptativa, Perfectiva e Preventiva;**
 - Corrigir erros; Adequar o sistema à uma nova realidade; e assim por diante
- **Reengenharia é um tipo de manutenção que normalmente implica ou deriva da reengenharia dos processos de negócios da organização usuária;**

18

■ Atividades guarda-chuva: transversais às demais etapas

- **Acompanhamento e controle do projeto de software;**
 - Coleta de Métricas;
- **Garantia de qualidade de software**
 - Revisões técnicas formais;
- **Gestão de configuração de software;**
- **Preparação e produção de documentos;**
- **Gestão de reutilização;**
- **Gestão de Risco**

19

Fases Genéricas do Processo de Desenvolvimento de Software

- | | |
|---------------------------------|----------------------------------|
| ■ Atividades Sequenciais | ■ Atividades em Paralelo |
| □ Análise; | □ Gerência de Processo; |
| □ Projeto de Alto Nível; | □ Garantia de Qualidade; |
| □ Projeto Detalhado; | □ Gestão de configuração; |
| □ Implementação; | □ Gestão de Reutilização; |
| □ Testes; | □ Gestão de Riscos; |
| □ Manutenção; | |

20

A Engenharia de Software inclui...

- Princípios, métodos e metodologias;
 - **Modularidade;**
 - **Diagramas de fluxo de dados;**
- Planejamento, estimativas, e gerência de projetos;
- Projeto de programas
 - **Algoritmos;**
 - **Estruturas de dados;**
- Ferramentas;
 - **Compiladores;**
 - **Geração de Testes;**
 - **Colaboração: email, CVS, Powerpoint;**

21

Engenharia de Software

- Estrutura:

[Guezzi-1994]



- Princípios:

- > Modularidade
- > Consistência
- > Completude
- > Confiabilidade
- > Corretude
- > Manutenibilidade
- > Reusabilidade
- > Eficiência
- > Abstração
- > Formalismo

22

Engenharia de Software

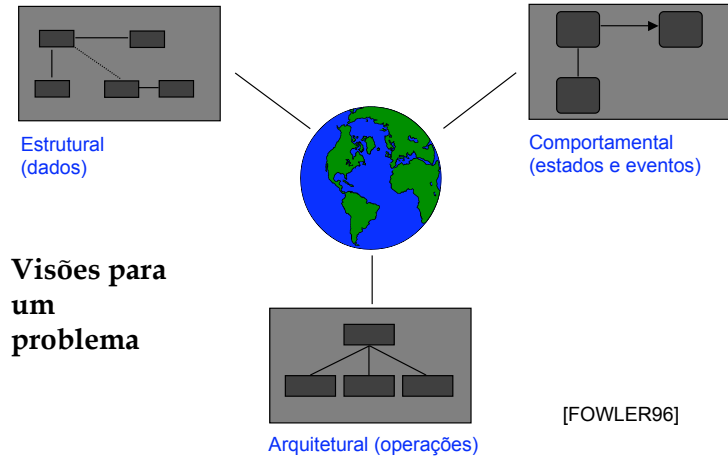
- Métodos:
 - **Modelos formais usados para:**
 - compreender
 - comunicar
 - **Exemplos: ER, DFD, DE, DD, SADT, NSD, DTE,**
- Metodologia ou Ciclo de vida:
 - **Conjunto de passos que visam a construção de software**
 - gerenciável, completo, abstração,
- Ferramentas
 - **Suportam uma metodologia ou um método;**

23

Métodos de Engenharia de Software

- Formalismo utilizado para representar o software em vários níveis de abstração;
- Precisão
 - **Ausência de ambigüidade**
- Apresentação:
 - **Textos**
 - **Diagramas**
- Representação de algum aspecto de software:
 - **Estrutural, Comportamental, Arquitetural, etc.**

24



25



Leitura opcional

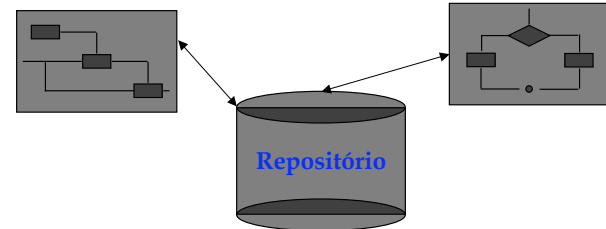
- “No Silver Bullet”, disponível em <http://www.ufpa.br/cdesouza>

27



Ferramentas

- Automatizam o processo de desenvolvimento de software
 - Editores, testadores, compiladores e gerenciadores de processo, gerência de configuração, etc.
 - Integração entre ferramentas;



26