

Introdução à Engenharia de Software

Estamos satisfeito com:

- a quantidade de falhas dos softwares que usamos?
- a dificuldade de fazer programas operarem em versões mais novas e aprimoradas de sistemas operacionais?
- o tempo que leva para produzir um software que garantidamente funcione?
- o esforço requerido para desenvolver software que possa ser distribuído sem que o cliente se sinta enganado?
- o custo de uso do software?
- . . .

Devemos nos Preocupar?

“We have become dangerously dependent on large software systems whose behavior is not well understood and which often fail in unpredicted ways.”

USA PITAC (President's Information Technology Advisory Committee), 1999

“Programming today is a race between software engineers striving to build bigger and better idiot-proof programs, and the Universe trying to produce bigger and better idiots. So far, the Universe is winning.”

Rich Cook

Problemas Famosos de Software

- Foguete Ariane 5, 1996
 - O foguete explodiu 40 segundos após a sua primeira decolagem.
 - Prejuízo de U\$ 500 milhões
 - Foi aproveitado um pacote de software de navegação do Ariane 4 que não tinha erros.
 - No módulo Sistema de Referencia Inercial uma conversão de valores de 64-bits para 16-bits causou um operando inválido que o interrompeu.
 - A falha no SRI levou o computador de bordo a modificar a trajetória do foguete levando à ruptura de juntas e isto causou a ativação da auto-destruição.

Problemas Famosos de Software

- Sistema de ambulâncias de Londres, 1992
 - Morte de pessoas que não foram socorridas em tempo.
 - Responsáveis contrataram uma empresa desconhecida cujo valor cobrado era menor que os cobrados pelas empresas de renome.
 - Colocaram o sistema no ar sem os devidos testes.
 - Não foi feita uma migração correta do sistema antigo para o novo.

Problemas Famosos de Software

- Máquina de radioterapia com software controlador, 1985-87
 - O controle de segurança feito pelo hardware em máquinas anteriores foi removido e passou a ser feito por software.
 - O software falhou na tarefa de, ao mesmo tempo, manter invariantes essenciais: o feixe de elétrons e o dispositivo que controla a concentração do feixe em níveis seguros.
 - Diversas mortes em decorrência de queimaduras.
 - O programador não tinha experiência em programação concorrente.

Problemas Famosos de Software

- Aeroporto de Denver, 1993
 - Sistema de triagem/controlado de bagagem
 - Atrasou a inauguração do aeroporto.
 - Custo do sistema: US\$ 193 milhões
 - Inauguração estava prevista para Out/1993.
 - Em Junho/1994 o sistema ainda não estava funcionando e causava prejuízos de US\$ 1,1 milhão/dia
 - No começo de 1995 um controle manual de bagagem foi instalado para que o aeroporto pudesse ser inaugurado (com atraso de mais de um ano)
 - Problema de planejamento e gerenciamento.

- Mas quantas vezes topamos com casos desse tipo na nossa vida profissional?
- E problemas menos “charmosos”, quantas vezes topamos com eles?

Software é "buguento"!

- Cerca de 40 a 50% dos programas postos em uso contém faltas não triviais
 - Boehm, B.W.; Basili, V.R.; "Software Defect Reduction Top 10 List"; IEEE Computer 34(1); Los Alamitos, CA: IEEE Computer Society; 2001; pags 135-137
- Como reduzir este percentual?
- Como reduzir os possíveis prejuízos?
- Como evitar a propagação de danos?
- Como reduzir o tempo médio para recuperar?
- Como reduzir o tempo médio para corrigir?

Projetos de Software Fracassam

- Muitos projetos de desenvolvimento de software demoram e custam muito mais do que o estimado.
- Muitos projetos de desenvolvimento de software nem chegam a conseguir finalizar o software desejado.
- Muitos softwares não servem para ser usados depois de desenvolvidos.

O que queremos?

- Ter a certeza de que estamos desenvolvendo economicamente software que possua qualidade e seja capaz de operar em ambientes reais.

O que precisamos?

Ter certeza de que estamos praticando

Engenharia de Software

O que é Engenharia de Software?

- É o emprego de processos, técnicas, ferramentas, matemática e disciplina para o desenvolvimento econômico e dentro dos prazos estabelecidos de software de forma que o software possua qualidade assegurada e seja capaz de operar nos ambientes reais a que se destina.
- A engenharia de software ainda é um campo novo e em desenvolvimento.

O que é Engenharia de Software?

- A IEEE Computer Society define engenharia de software como: "(1) The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software. (2) The study of approaches as in (1)."
- IEEE Computer Society
 - Uma das principais organizações mundiais de profissionais em computação.
 - Dedicada a promover avanços na teoria e aplicação de computação e TI.
 - Promove conferências, periódicos, revistas, padrões, guias, livros, certificações.

Aspecto Histórico

- 1968 Conferência da OTAN, Garmisch
- Objetivo: resolver a “Crise do Software”.
- Software é constantemente entregue
 - Atrasado
 - Estimativas erradas, baixa produtividade.
 - Com orçamento estourado
 - Duplicação de esforços, alto custo de manutenção.
 - Com falhas residuais
- Custo do hardware decrescente e custo do software em ascensão

Software Engineering Body of Knowledge (SWEBOK)

- Corpus de Conhecimento sobre Engenharia de Software
 - Mantido pela IEEE Computer Society
 - www.swebok.org

Software Engineering Body of Knowledge (SWEBOK)

- The SWEBOK Knowledge Areas (KAs)
 - Requisitos
 - Design
 - Construção
 - Teste
 - Manutenção
 - Gerenciamento de Configurações
 - Gerenciamento de Projeto (Gerenciamento de Engenharia)
 - Engenharia de Processo
 - Ferramentas e Métodos
 - Qualidade de Software
- Novas KAs em processo de definição
 - Medição
 - Segurança