Engenharia de Software

Alexandre Vasconcelos, André Santos,
Augusto Sampaio, Hermano Moura,
Paulo Borba

Introdução à Engenharia de Software

Paulo Borba

Objetivo

Depois desta aula você terá uma noção geral do que é a engenharia de software e dos seus objetivos e conceitos básicos relacionados.

Introdução à Engenharia de Software

Paulo Borba

Leitura prévia essencial

- Capítulo 1 do livro *Software Engineering—A Practioner's Approach* (de Roger Pressman)

O que é a Engenharia de Software?

Estudo ou aplicação de abordagens sistemáticas, econômicas e quantificáveis para o desenvolvimento, operação e manutenção de software de qualidade.

Objetivos da Engenharia de Software

- Qualidade de software
- Produtividade no desenvolvimento, operação e manutenção de software
- Qualidade *versus* Produtividade
- Permitir que profissionais tenham controle sobre o desenvolvimento de software dentro de custos, prazos e níveis de qualidade desejados

Qualidade de Software (para o Varejo)

- Correto
  - A loja não pode deixar de cobrar por produtos comprados pelo consumidor
- Robusto e altamente disponível
  - A loja não pode parar de vender
- Eficiente
  - O consumidor não pode esperar
  - A empresa quer investir pouco em recursos computacionais (CPU, memória, rede)

Qualidade de Software (para o Varejo)

- Amigável e fácil de usar
  - A empresa quer investir pouco em treinamento
- Altamente extensível e adaptável
  - A empresa tem sempre novos requisitos (para ontem!)
  - A empresa quer o software customizado do seu jeito (interface, teclado, idioma, moeda, etc.)
- Reusável
  - Várias empresas precisam usar partes de um mesmo sistema

Qualidade de Software (para o Varejo)

- Aberto, compatível, de fácil integração com outros sistemas
  - A empresa já tem controle de estoque, fidelização, etc.
- Portável e independente de plataforma (hw e sw)
  - A empresa opta por uma determinada plataforma
- Baixo custo de instalação e atualização
  - A empresa tem um grande número de PDVs

Produtividade

- Custo de desenvolvimento reduzido
  - A empresa consumidora quer investir pouco em software
  - A empresa produtora tem que oferecer “software barato”
- Tempo de desenvolvimento reduzido
  - Suporte rápido às necessidades do mercado

“Software Barato”

Nem tanto resultado de baixos custos de desenvolvimento, mas principalmente da distribuição dos custos entre vários clientes.

Reuso, extensibilidade e adaptabilidade são essenciais para viabilizar tal distribuição.

Relevância da Engenharia de Software

Qualidade de software e produtividade garantem:

- Disponibilidade de serviços essenciais
- Segurança de pessoas
- Competitividade das empresas
  - Produtores
  - Consumidores

Mas, na realidade, temos a Crise de Software...

- 25% dos projetos são cancelados
- o tempo de desenvolvimento é bem maior do que o estimado
- 75% dos sistemas não funcionam como planejado
- a manutenção e reutilização são difíceis e custosas
- os problemas são proporcionais a complexidade dos sistemas

Causas da Crise de Software

- Essências
  - Complexidade dos sistemas
  - Dificuldade de formalização
- Acidentes
  - Má qualidade dos métodos, linguagens, ferramentas, processos, e modelos de ciclo de vida
  - Falta de qualificação técnica

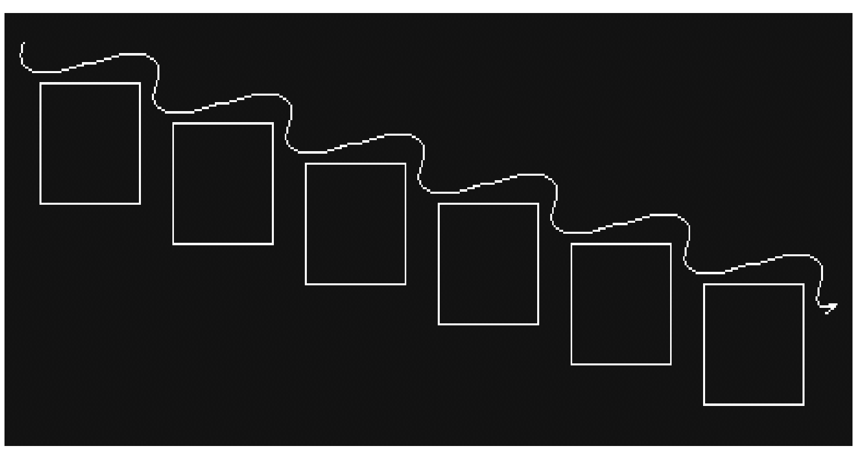
Elementos e Atividades da Engenharia de Software

- Elementos
  - Modelos do ciclo de vida do software
  - Linguagens
  - Métodos
  - Ferramentas
  - Processos
- Atividades
  - Modelagem do negócio
  - Elicitação de requisitos
  - Análise e Projeto
  - Implementação
  - Testes
  - Distribuição
  - Planejamento
  - Gerenciamento
  - Gerência de Configuração e Mudanças
  - Manutenção

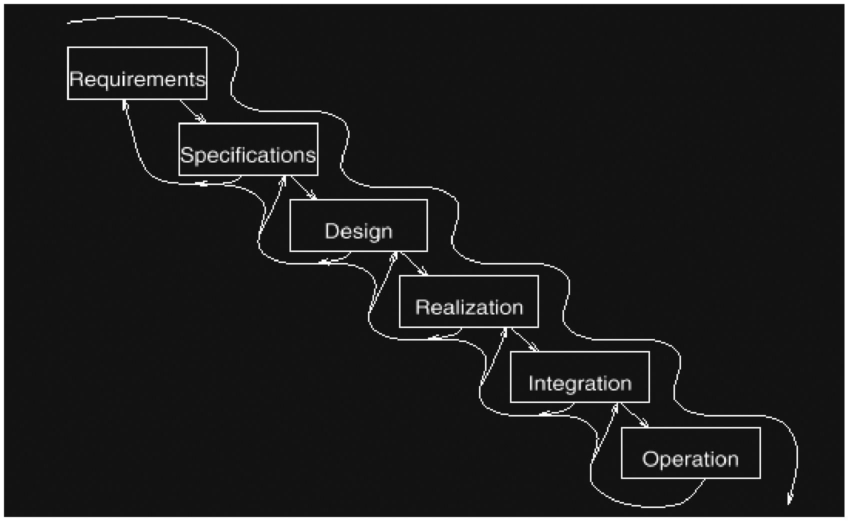
Modelos do Ciclo de Vida de Software

- Cascata
- Espiral
- Iterativo do RUP
- Prototipagem evolucionária
- Força bruta, gambiarra, *hacking, ...*

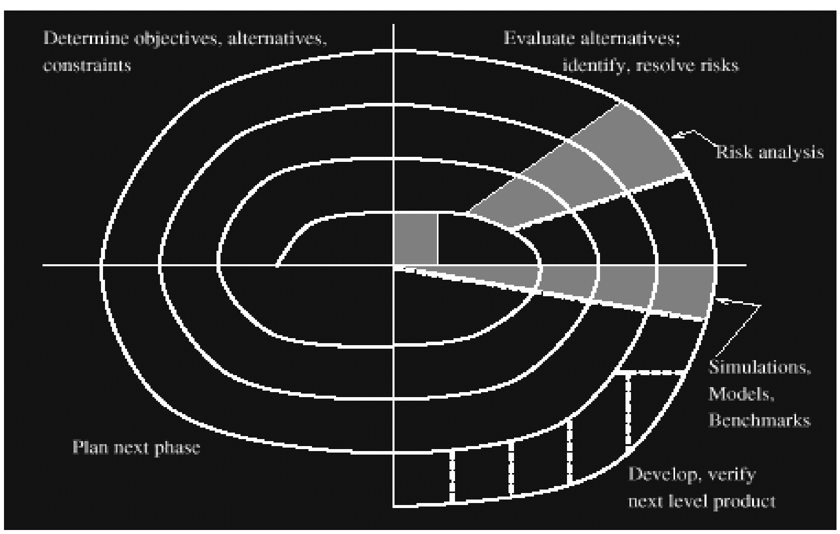
Modelo Cascata



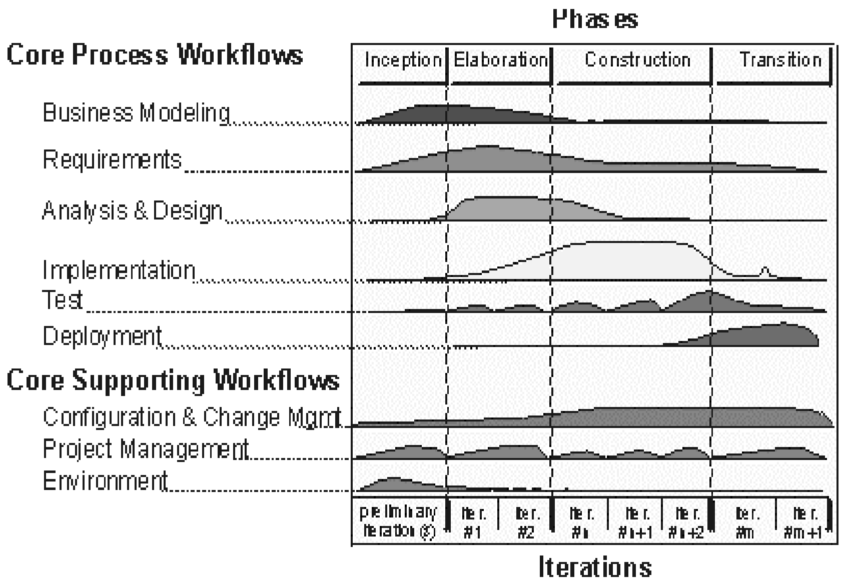
Modelo Cascata na Prática



Modelo Espiral



Modelo Iterativo do RUP



Linguagem

- Notação com sintaxe e semântica bem definidas
  - com representação gráfica ou textual
- Usada para descrever os artefatos gerados durante o desenvolvimento de software
- Exemplos: UML, Java

Método

- Descrição sistemática de *como* deve-se realizar uma determinada atividade ou tarefa
- A descrição é normalmente feita através de padrões e guias
- Metodologia
  - conjunto de métodos **(+ processo)**
- Exemplos: Booch, BON, Pim

Ferramenta

- Provê suporte computacional a um determinado método
- Ambiente de desenvolvimento: conjunto de ferramentas integradas (CASE)
- Exemplos: Rational Rose, JBuilder

Processo

- Conjunto de atividades
  - bem definidas
  - com responsáveis
  - com artefatos de entrada e saída
  - com dependências entre as mesmas e ordem de execução
  - com modelo de ciclo de vida

Introdução à Engenharia de Software

Paulo Borba

Resumo

- Engenharia de software: objetivos e relevância
- Produtividade e qualidade de software
- Crise de software
- Elementos e atividades da engenharia de software

Introdução à Engenharia de Software

Paulo Borba

Leitura adicional

- Daniel M. Berry. *Myths and Realities in Software Development*.
- W. Wayt Gibbs. *Software's chronic crisis*. Scientific American, September 1994.
- Alan Joch. *How software doesn't work*. Byte, December 1995.
- IEEE e ACM. *Software Engineering Code of Ethics and Professional Practice*.